

Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

BAKALÁRSKA PRÁCA



Attila Ulman

Vznik inteligentního chování u hmyzích společenstev

Katedra teoretické informatiky a matematické logiky

Vedúci bakalárskej práce: RNDr. Pavel Surynek
Študijný program: Informatika, Obecná informatika

2007

Rád by som poďakoval každému, kto mi bol pri písaní bakalárskej práci nápomocný, bez koho by vznik práce nebol možný. Predovšetkým ďakujem môjmu vedúcemu, RNDr. Pavlovi Surynkovi, za jeho cenné pripomienky a rady, a za čas, ktorý mi počas vypracovania tejto práce venoval a taktiež svojej rodine za stálu podporu.

Prehlasujem, že som svoju bakalársku prácu napísal samostatne a výhradne s použitím citovaných prameňov. Súhlasím so zapožičiavaním práce a jej zverejňovaním.

V Prahe dňa 1. augusta 2007

Attila Ulman

Obsah

1 Úvod	5
1.1 Motivácia.....	5
1.2 Ciele práce.....	7
2 Príroda – inšpirácia	8
2.1 Mravce – tajná sila prírody	8
2.2 Sociálny hmyz.....	9
3 Umelý život	15
3.1 Inteligencia.....	15
3.2 Čo je umelý život?	17
3.3 Evolúcia v počítači – „in silico“.....	20
4 Prístupy riešenia problematiky	22
4.1 Existujúce aplikácie, možnosti riešenia	22
4.2 Vlastné vybrané postupy riešenia	25
5 Výsledky pokusov	32
5.1 Experimenty	32
5.2 Vyhodnotenie pokusov	33
6 Záver	37
6.1 Konklúzia a myšlienky budúcej práce	37
Literatúra	40
Dodatok	42
Prílohy	43

Názov práce: Vznik inteligentního chování u hmyzích společenstev
Autor: Attila Ulman
Katedra (ústav): Katedra teoretické informatiky a matematické logiky
Vedúci bakalárskej práce: RNDr. Pavel Surynek
e-mail vedúceho: Pavel.Surynek@mff.cuni.cz

Abstrakt:

Práca sa zaoberá skúmaním a simuláciou rozsiahleho spoločenstva jednoduchých agentov – mravcov. Títo relatívne jednoduchí jedinci v spojení dokážu dosiahnuť oveľa zložitejšie ciele, ktoré viacnásobne prevyšujú ich individuálne schopnosti. Sú schopní kolektívne riešiť úlohy, ktoré by jednotlivci nemali šancu zvládnuť a to bez existencie nejakej riadiacej autority a dopredu daných fixných štruktúr, napriek prítomnosti silného vnútorného šumu. Aplikáciou jednoduchých pravidiel vzniká komplexné chovanie celku. Zaujímavý je najmä ich primárny životný cieľ – hľadanie potravy – práve tu sa najviac prejavuje ich kooperácia a inteligencia celého systému ako „superorganizmu“. Pochopenie emergentných schopností mravčích kolónií môže byť prínosné pri štúdiu kolektívneho chovania vyspelejších spoločenstiev.

Zámerom práce je skúmať chovanie sociálneho hmyzu a pokúsiť sa vytvoriť simuláciu „in silico“ ktorá je čo najbližšie k realite a čo najviac zachytuje inteligentnú kooperáciu jedincov.

Kľúčové slová: umelý život, sociálny hmyz, počítačová simulácia

Title: Emergence of Intelligent Behaviour of Social Insects
Author: Attila Ulman
Department: Department of Theoretical Computer Science and Mathematical Logic
Supervisor: RNDr. Pavel Surynek
Supervisor's e-mail address: Pavel.Surynek@mff.cuni.cz

Abstract:

The aim of this work is to study and simulate the extensive community of simple agents – ants. These relatively simple individuals are – by working together – capable of achieving complex goals that are far beyond the abilities of a single one of them. They are able to collectively solve problems, exceeding the capability of a single creature in the absence of any kind of a teaching authority or a goal given in forward. By applying simple rules, a complex behavior of the collective can emerge. The most interesting part is the primary life goal: the search for food. This is where the cooperation and the acting of the system as a whole super organism can be observed. With the help of understanding the emergent capabilities of ant colonies, the collective behavior of more advanced societies could be examined.

The goal of our work is to study the behavior of social insects and an attempt to create an „in silico“ simulation that is realistic, and shows the intelligent cooperation of individuals.

Keywords: artificial life, social insects, computer simulation

Kapitola 1

Úvod

1.1 Motivácia

Keď sa bližšie pozrieme na vývoj prírodných vied, vidíme, že nové nástroje, ktoré vznikajú pokrokom techniky, nesmiernym spôsobom urýchľujú napredovanie tej ktorej oblasti vedy. Otvárajú vedcom dovtedy nepreskúmané, uzavreté teritória, alebo prinášajú také nové metódy, pomocou ktorých sa získavanie znalostí, alebo prenášanie teórií do praxe významnou mierou zrýchli. Takýmto významným nástrojom z pohľadu histórie vedy je možné považovať napr. v oblasti biológie mikroskop, v astronómii teleskop, alebo vo fyzike urýchľovač častíc, no do tejto prominentnej spoločnosti určite musíme zaradiť aj počítač, ktorého význam nie je možné obmedziť na jednu alebo niekoľko oblastí vedy, nakoľko jeho prítomnosť značne vplýva na prírodné vedy ako celok. Je tomu tak i v tých oblastiach vedy, ktoré skúmajú komplexné adaptívne systémy, resp. javy, pri ktorých sa systém skladá z veľkého počtu jednotlivito relatívne jednoduchých prvkov a ich vzájomného pôsobenia, má nové vlastnosti, chová sa ako nezávislý celok. Adaptuje sa na zmeny prostredia a vykazuje také emergentné vlastnosti, ktoré nie je možné odvodiť od vlastností jednotlivých komponentov [1]. Ozajstné skúmanie a analýza týchto systémov mohli prísť až s vývojom počítačových vied, ako tvrdí Mitchell [2].

Hore uvedenú definíciu skoro bez výnimky spĺňajú všetky biologické systémy, od základnej stavebnej jednotky živých organizmov – bunky, po imunitný systém, od ľudskej inteligencie, po biosféru, ba dokonca aj ľudská spoločnosť podlieha týmto zákonitostiam [3]. Nemožno sa preto čudovať, že s rozšírením výpočtovej techniky sa i vedný odbor biológie zameral na nové vedecké smery, ktoré si vytýčili za cieľ počítačové modelovanie biologických systémov v daných oblastiach. Skúmanie určitých súvislostí a pomerov spomínaných systémov tradičnými postupmi bádania by bolo často priveľmi zložité, ba dokonca niekedy aj nemožné. V takýchto prípadoch dáva počítačové modelovanie novú šancu pre lepšie porozumenie daných javov. V posledných rokoch je viditeľná tendencia poukazujúca na to, že popri tradičných zbraniach výskumného arzenálu, ako sú matematické teórie a experimentálne metódy, sa začínajú čoraz intenzívnejšie udomáčať i počítačové simulácie [2].

Pri pohľade na posledných 50 rokov výskumu komplexných adaptívnych systémov sa ukazujú 4 prekrývajúce sa obdobia. Ich míľnikom je objavenie sa nových teórií a následne ich vplyv na názor, uhol pohľadu výskumníkov. Päťdesiate až šesťdesiate roky sa niesli v znamení kybernetiky, od polovice sedemdesiatych rokov sa stáva smerodajným prúdom výskumov teória chaosu a nelineárna

dynamika. Nasledujúce desaťročie prinieslo vypracovanie a postupne i širokú aplikáciu teórie neurónových sietí, ktoré i dnes vykazujú dynamický rozvoj. Na tieto sa koncom osemdesiatych rokov čiastočne nadviazalo skúmanie „umelého života“ a „autonómnych agentov“. V tejto tendencii vývoja sa prejavujú 2 aspekty: na jednej strane vidíme, že vzťah medzi biológiou a počítačovými vedami nie je jednosmerný, nakoľko nie len biologické výskumy zužitkovávajú potenciál výpočtovej techniky, ale aj výsledky získané pri pozorovaní biologických systémov majú čoraz väčší vplyv na informatiku. Spomeňme len umelé neurónové siete a ich významné aplikácie, ktoré už dnes majú za sebou peknú kariéru [4], paralelné počítače [5], alebo rastúce využitie genetických algoritmov a evolučného programovania [4]. Na druhej strane si môžeme všimnúť aj to, že dnes už počítač nie je len prostriedkom výskumu komplexných adaptívnych systémov, ale čoraz väčšmi sa stáva novým prostredím, ktoré pre niektoré typy týchto systémov slúži ako životný priestor [6, 7].

„Štúdiom chovania sociálneho hmyzu sa môžeme veľa naučiť,“ hovorí známy sociobiológ Bert Hoelldobler. V osemdesiatych rokoch minulého storočia napísal článok s názvom: Territorial Strategies in Ant Societies. Bol ohromený, z koľkých miest dostal požiadavky na dotlač – z Kennedy School of Government v Harvarde, z Pentagonu. Veľmi sa čudoval, prečo sa zaujímajú o niečo také. Výskumník z Harvardu mu povedal: „Zaujíma nás, ako sa počas evolúcie optimalizovali teritoriálne stratégie.“

(Kapitola 2.1 Mravce – tajná sila prírody)

Motivovaný predchádzajúcimi podnetmi sa v práci zaoberám skúmaním a simuláciou rozsiahleho spoločenstva jednoduchých agentov – mravcov. Títo relatívne jednoduchí jedinci dokážu v spojení dosiahnuť oveľa zložitejšie ciele, ktoré viacnásobne prevýšia ich individuálne schopnosti. Sú schopní kolektívne riešiť úlohy, ktoré by jednotlivo nemali šancu zvládnuť a to bez existencie nejakej riadiacej authority a dopredu daných fixných štruktúr. Aplikáciou jednoduchých pravidiel vzniká komplexné chovanie celku. Zaujímavý je najmä ich primárny životný cieľ – hľadanie potravy – práve tu sa najviac prejavuje ich kooperácia a inteligencia celého systému ako „superorganizmu“, „inteligentnej entity“. Kolónia si veľmi rýchlo vie nájsť cestu k potrave pomocou feromónových značiek. Pri pohybe od mraveniska k zdroju potravy a späť každý mravec ukladá chemickú látku zvanú feromón. Ostatné mravce sú schopné detekovať feromón a aj rozlišovať jeho rôzne koncentrácie a podľa najvyššej si vyberajú cestu k potrave. Čím viac mravcov používa danú cestu, tým väčšia bude jej koncentrácia feromónu, kým ostatné nepoužívané cesty pomaly vyprchávajú. Žiadna prekážka nedokáže tomuto putovaniu zabrániť, mravce si vždy nájdu ďalšiu najlepšiu cestu. Týmto vlastne vytvárajú dynamicky sa meniacu štruktúru medzi nimi (mraveniskom) a potravou, ktorá je schopná prispôbovať sa meniacim podmienkam. Vlastnosti tohto typu sa s obľubou čoraz častejšie používajú pre optimalizačné úlohy (Ant Colony Optimization) a pre aplikácie v sieťovom prostredí (routing).

Dá sa povedať, že mravce komunikujú cez prostredie: Aktivita jedincov má za výsledok feromónovú stopu, ktorú mravce nasledujú, čiže prostredie vplýva na ich aktivitu. Týmto vzniká nepriama komunikácia medzi jedincami. Samozrejme je tu ešte mnoho vplyvov, ako napr. zdroj svetla, vlhkosť, prekážky, samotné vlastnosti jedince, atď.
(podrobnejší popis z hľadiska prírodných súvislostí je v Kapitole 2)

1.2 Ciele práce

Ide o výskumnú prácu, cieľom ktorej je skúmať chovanie sociálneho hmyzu, jeho vznik, pozadie a súvislosti a pokúsiť sa vytvoriť simuláciu „in silico“ ktorá je čo najbližšie k realite a čo najviac zachycuje inteligentnú kooperáciu jedincov. Pokúsiť sa vytvoriť inteligentné chovanie kolónie a dosiahnuť zaujímavé teoretické výsledky. Pomocou simulácie pochopiť, ako môžu napr. mravce kolektívne uspieť, prekonať tak svoje individuálne obmedzené schopnosti. Taktiež pochopiť emergentné schopnosti mravčích kolónií, ktoré môžu byť nástrojom štúdia kolektívneho chovania vyspelejších spoločenskíev.

Ďalším významným cieľom je porovnať rôzne prístupy riadenia jedincov a skúmať, ktorá metóda je najviac realistická resp. najefektívnejšia (čo sa týka putovania za potravou). Pri dokonalosti prírody tieto dva aspekty majú k sebe veľmi blízko. Skúmať rôzne analógie s reálnou prírodou a tiež vplyv rôznych zmien vstupných parametrov simulácie. Vyvodit' závery jednak z porovnania prístupov riadenia, algoritmov ako aj z rôznych zmien parametrov, prostredia, vlastností celého systému. Dôraz sa kladie na čo najvernejšiu podobu prírodných javov a na samotné vytvorenie inteligentného chovania.

Simulačný systém – samotná aplikácia (SocInCo) – umožňuje študovať vznik inteligentného chovania hmyzích spoločenskíev a porovnávať rôzne prístupy. Jej súčasťou je vizualizácia simulácie, vyhodnocovanie simulácie, porovnávanie, grafické znázornenie parametrov, úspešnosti kolónie, ich uloženie, zadávanie parametrov simulácie, prostredia. Obsahuje dve metódy riadenia jedincov: konečný automat a neurónovú sieť v kombinácii s genetickými algoritmami.

Kapitola 2

Príroda – inšpirácia

2.1 Mravce – tajná sila prírody

Mravce – záhadné a tajuplné spoločenstvo. Dlhé roky sa nepodarilo nabúrať ich tajné kódy, pochopiť zložitý systém komunikácie týchto priam mimozemských tvorov, ktoré podivuhodne udržia stonásobok svojej váhy, alebo sú schopné stáť dolu hlavou na skle. Ani dnes sa však ešte nepodarilo rozlúštiť všetky ich záhady [D].

Sociobiológ Bert Hoelldobler, medzinárodne uznávaný výskumník a autorita oblasti zaoberajúcej sa jednou z najúspešnejších životných foriem na Zemi – mravcami – už 60 rokov študuje tento ohromne fungujúci decentralizovaný systém.

Mravce sa v skutočnosti vyskytujú v každom pozemskom prostredí, od tundry cez dažďové pralesy až po púšť, na každom kontinente, okrem Antarktídy. I keď sú drobné, ich celkový objem na planéte súperí s ľudským. Odhaduje sa, že veľkosť ich globálnej populácie je okolo 10^{16} . Napr. v oblasti Amazonky, ktorá je bohatá na život, predstavujú tretinu celej živočíšnej biomasy.

Napriek tomu, že väčšina ľudí žijúcich v meste považuje mravce za škodcov, hrajú kritickú rolu v mnohých ekosystémoch ako hlavní predátori iného hmyzu, významní spracovávatelia pôdy aj ako hlavní distribútori výživných látok a semien rastlín. Tisíce druhov mravcov charakterizuje vysoká špecializácia, jedia všetko od výlučkov listových vošiek cez rastliny, živočíchy, až po huby, ktoré aj sami pestujú.

Hoelldoblerov záujem o mravce prekračuje hranice ich ekologického významu a zameriava sa na hlavný dôvod ich úspechu, na ich vysoko vyvinuté sociálne chovanie a sociálnu zložitosť.

Mravce spolu s ich príbuznými, včelami a osami, patrili počas evolúcie medzi prvé organizmy, u ktorých sa pred viac ako sto miliónmi rokov vyvinulo pravé sociálne chovanie.

Práve pre tento fakt je sociálny hmyz dôležitým objektom štúdie nie len prírodovedcov, ekológov a entomologistov, ale i výskumníkov zaoberajúcich sa sociálnym chovaním, biológiou sociálnych systémov.



Obr. 1 Snímok z dokumentu [D]

„Študovaním chovania sociálneho hmyzu sa môžeme veľa naučiť,“ hovorí Hoelldobler. V osemdesiatych rokoch minulého storočia napísal článok s názvom: Territorial Strategies in Ant Societies. Bol ohromený, z koľkých miest dostal požiadavky na dotlač – z Kennedy School of Government v Harvarde, z Pentagonu. Veľmi sa čudoval, prečo sa zaujímajú o niečo také. Výskumník z Harvardu mu povedal: „Zaujíma nás, ako sa počas evolúcie optimalizovali teritoriálne stratégie.“

„Mravce sú veľmi agresívne spoločenstvo. Keby disponovali atómovou bombou, naša planéta by už neexistovala. Nezodpovedá to úplne pravde, lebo spôsob, akým vedú vojny, závisí vo veľkej miere na distribúcií zdrojov – v závislosti na nich si vytvárajú rôzne územné stratégie. Žiadna z kolónií neinvestuje do ochrany územia viac síl, ako môže získať späť z jej zdrojov. Je to vzťah nákladov a výnosov.“ – vysvetľuje Hoelldobler.

Genetické základy niektorých sociálnych charakteristík sú totožné u mravcov aj u ľudí. Napriek tomu, že sa vyvíjali nezávisle od seba, základné sociálne organizácie ľudí a spoločenstvo mravcov sú podobné.

„Nechcem byť ten, kto nazve mravce malou ľudskou spoločnosťou – sú úplne odlišné,“ hovorí Hoelldobler. „Ale to, čo sa môžeme naučiť študovaním biodiverzity živočíšnych spoločenstiev sú určité princípy, ktoré sa u mravcov objavujú. Je to napríklad deľba práce, sociálna hierarchia, sociálna sieť, systém komunikácie, rozpoznávanie príbuzných, atď.“ [D]

2.2 Sociálny hmyz

Predošlá podkapitola len potvrdzuje fakt, že sociálny hmyz zaujíma kľúčovú úlohu v oblasti štúdia umelého života vďaka jednoduchosti jedinca v spojení so zložitým kolektívnym správaním. Tento spoločne dokáže dosiahnuť oveľa zložitejšie ciele, ktoré viacnásobne prevýšia jeho individuálne schopnosti. Je schopný tímovo riešiť úlohy, ktoré by jednotlivo nemal šancu zvládnuť a to bez existencie nejakej riadiacej autority a dopredu daných pevných štruktúr. Aplikáciou prostých pravidiel vzniká zložitá chovanie celku.

Vo vlastných pomeroch rieši mnohé úlohy veľmi podobné úlohám v ľudských spoločnostiach.

Niekoľko pojmov oboru [8, 9, D]:

kasta	určité špecializované spoločenstvo sociálneho hmyzu, ktoré má rôzne úlohy, niekedy aj morfológiu
eusociálny systém	sociálny systém s rodičovskou starostlivosťou o mladé, prelínajúcimi sa generáciami a reprodukčnou deľbou práce – skutočná spoločnosť
homeostáza	uchovanie funkčného a nemenného stavu v organizme alebo superorganizme
polyetizmus	diferencie v chovaní medzi rôznymi kastami, deľba práce, špecializácia

polymorfizmus	úplná vzhľadová odlišnosť členov v závislosti na rozdieloch prostredia
sociálny hmyz	v kolóniách kooperatívne žijúci hmyz, s deľbou práce medzi kastami (patria medzi ne: termity, mravce, včely a niektoré osy)
superorganizmus	kolónia sociálneho hmyzu charakterizovaná ako mnohobunkový živočích, jednotliví členovia kolónie tvoria zložky organizmu

Znaky eusociálneho hmyzu

- starostlivosť rodičov o mladé jedince (bez nej by nemohli prežiť)
- prelinanie sa generácií (dôležité pre vlastnosť č. 1)
- reprodukčná deľba práce (napr. samičky, ktoré kladú vajčka a ostatné, patriace do iných kást.)

Emergentné chovanie

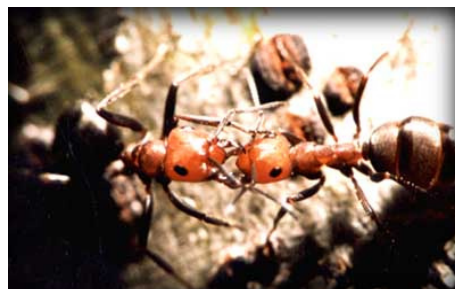
Je to chovanie, ktoré sa pozorovateľovi javí ako komplexné alebo zmysluplné, ale v skutočnosti je samovoľne odvodené od viacerých jednoduchých pravidiel. Nízkoúrovňové entity vykonávajúce jednoduché úkony, nedisponujú žiadnou predstavou o väčšom obraze, majú prehľad len o sebe či možno o niekoľkých svojich susedoch. Emergentné chovanie sa vyskytuje skoro všade, od tiahnutia množstva rýb a hľadania potravy mravcov, až po prúdy dopravných vozidiel, vývin živočíšnych spoločností a štruktúru ekonómie či finančných trhov. Jedným z najznámejších príkladov je zhľukovanie sa vtáčích krdľov (flocking) objavený Craigmom Reynoldsom. Vyvinul „Boidov“, ktorých chovanie je prekvapivo podobné zhľukovaniu vtákov (stačí im k tomu nasledovať tri jednoduché pravidlá: separácia, usmernenie, kohézia).

Mravec ako jednotlivec nemá globálne vedomosti o úlohe ktorú vykonáva. Akcie mravca sú založené na lokálnych rozhodnutiach a sú väčšinou nepredikovateľné. Inteligentné chovanie sa prirodzene objavuje ako konsekvencia sebaorganizácie a nepriamej komunikácie medzi jedincami – emergentné chovanie.

Altruizmus

Je to vôľa zriecť sa osobných záujmov v prospech ostatných, nesebecká práca pre dobro spoločenstva.

Ak sa zamyslíme nad chovaním pracujúceho mravca, dôjdeme k záveru, že je to evolučný nezmysel. Prečo obetuje jedinec šancu prenesenia vlastných génov do ďalšej generácie, len aby pomohol inému jedincovi?



Obr. 2 Navzájom sa kŕmiace mravce

Klasický darwinovský pohľad si nevedel poradiť s týmto javom, až kým neprišiel Brit William Donald Hamilton, ktorý v roku 1963 publikoval myšlienku

sociálneho správania (Hamiltonova teória rozšírenej reprodukčnej zdatnosti a príbuzenského výberu), táto sa týkala hmyzov radu Hymenoptera (patria sem včely, osy ale i mravce). Upriamil pozornosť na to, že tento druh hmyzu je geneticky predurčený k sociálnemu správaniu, a to kvôli spôsobu, akým sa dedí pohlavie (haplodiploidne).

Rozmnožovanie

Podľa ľudských pomerov je plodnosť úspešných kráľovien ohromujúca. Kráľovné listnatých mravcov v Južnej a Strednej Amerike splodia 150 miliónov robotníč, z ktorých 2 až 3 milióny žijú súčasne.

Mravce žijú od 8 do 10 týždňov, prechádzajú štyrmi obdobiami života: vajíčko, larva, kukla a vyvinutý jedinec. Robotníci sú sterilné samičky, ktoré sa starajú o hniezdo. Väčší jedinci (vojaci) ochraňujú kolóniu.

Kráľovná sa behom svojho života pári len počas veľmi krátkeho úseku, ale samčie pohlavné bunky uskladňuje a rozdeľuje na celý svoj život. Kráľovná rozhoduje o tom, či dané vajíčko oplodní samčím pohlavným bunkami, alebo nie. Keď tak spraví, vyvinie sa diploidná robotnica, ak nie, tak haploidný samec (tzv. haplodiploidné rozmnožovanie). Kráľovná je len špeciálne kŕmená samička.

Spoločenstvo

Sociálne správanie hmyzu je väčšinou ojedinelé, spájajú, s inými členmi svojho druhu sa kontaktujú iba v krátkom období párenia, alebo pri iných príležitostných udalostiach. Sú medzi nimi i druhy, ktoré sa úplne vzdali aj párenia a rozmnožujú sa nepohlavne. Poznáme len niekoľko skupín, ktoré sú skutočne sociálne. Všetky termity (Isoptera), niektoré blankokrídlovce (všetky mravce, niekoľko skupín včiel a najmenej jeden druh ôs). Skutočný sociálny hmyz, predovšetkým mravce a termity, sú dominantné ekologické zoskupenia.

Je mnoho výhod sociálneho spôsobu života hmyzu: zvýšená produktivita kolónie, skupinové obranné mechanizmy, výstražné signály, získanie a transport potravy, spoločná výstavba hniezda, rodičovská starostlivosť o mladé jedince, atď. Na druhej strane sú aj určité nevýhody: väčšie škody pri chorobách, parazitizmu a útokoch. Pre udržanie spoločenstva musia omnoho úspešnejšie prežiť a rozmnožovať sa ako samotárski jednotlivci.

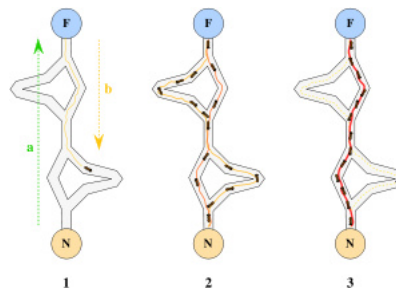
Protipól predchádzajúceho spôsobu je samotársky typ. Medzi jeho výhody je možné zaradiť: ľahšiu možnosť úkrytu pred predátormi, absenciu potreby súťažiť s ostatnými jedincami daného druhu, život v menších priestoroch, využitie menších zdrojov potravy, atď. Najväčšia nevýhoda je určite nedostatok sociálnych benefícií.

Komunikácia

Mravce sa dorozumievajú zväčša chemickými signálmi. V prípade, ak jedinec natrafí na veľký úlovok, na ktorý sám nestačí, vypustí poplašný feromón, ktorým v určitom okolí vzruší ostatné robotnice, čím ich privolá na miesto nálezu a spoločne ľahšie zmôžu korisť.

Najdôležitejším typom komunikácie, ktorá vedie k nájdeniu potravy, k vytvoreniu optimálnej cesty medzi mraveniskom a potravou, ako aj k jej prípadnej adaptácii je komunikácia prostredníctvom feromónových ciest.

Mravce formujú a udržiavajú líniu k zdroju potravy vylučovaním stopy feromónu, chemikálie, na ktorú sú ostatní členovia daného



Obr. 3 Princíp vzniku optimálnej cesty

druhu veľmi citliví, je základom určitej indirektnej komunikácie medzi nimi. Mravce hľadajúce potravu zanechávajú takú pachovú stopu, ktorá stimuluje stigmergetické chovanie ostatných mravcov k jej nasledovaniu a zároveň aj k jej posilneniu. Tento autokatalytický proces pokračuje dovtedy, pokiaľ sa stopa medzi kolóniou mravcov a zdrojom potravy nestabilizuje a tak umožňuje kolónii rýchle nájdenie najkratšej cesty. Mravce, ktoré sa vrátia najskôr, sú väčšinou tie, ktoré boli na kratších okruhoch, čiže tieto okruhy budú prvé dvojito označené feromónom (raz v každom smere). Teda ostatné mravce budú viac priťahované touto cestou ako dlhšími, menej krát použitými, čo znamená, že bude ešte silnejšie označená.

Čoskoro si preto drvivá väčšina mravcov zvolí túto cestu. Čo sa však stane v prípade, že mravec sa vráti skôr z dlhšej cesty, označujúc ju tak za najsilnejšiu? Počítačové simulácie ukazujú, že tento problém sa eliminuje, keď sa feromón pomaly stráca a vyparuje. Toto spôsobuje ťažšie udržanie stabilných feromónových stôp na dlhších cestách. Pre mravcov nie je cieľom vytvorenie „cesty“, ktorá je najkratšia, ale dovezenie potravy do mraveniska, pričom veľká väčšina takto vytvorených ciest je vysoko optimalizovaná.

Štúdiom tejto schopnosti mravcov sa otvorila pre výskumníkov možnosť k vytvoreniu softwarových agentov schopných riešiť komplexné informačné problémy, ako napr. znovusmerovanie v zaneprázdnených komunikačných sieťach.

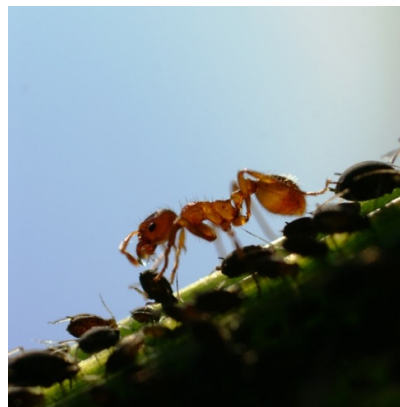
No je naivné myslieť si, že feromón je jediný prostriedok pri navigácii mravcov. V jednom ďalšom experimente bolo dokázané, že feromónové stopy nie sú polarizované, čo si do tej doby mnohí mysleli. Experiment obsahoval mravenisko, zdroj potravy a otočnú časť prostredia medzi mraveniskom a potravou. Po sformovaní feromónovej stopy bol vypustený z mraveniska mravec. Počas nasledovania stopy bol otočený o 180 stupňov. Keby pokračoval v ceste ako si to každý predstavoval, skončil by zas v mravenisku, no on prekvapujúco zmenil svoj smer a došiel do pôvodného miesta určenia. Tento experiment poukázal na to, že sa mravce spoliehajú aj na iné vnemy pri navigácii, ako je napr. pozícia slnka (alebo dosť silný svetelný zdroj), gravitácia, naklonenie či referenčné objekty.

Zaujímavé momenty, možná inšpirácia do budúcnosti

Nasledujúce fakty [8, D] by si zaslúžili hlbší výskum, nakoľko môžu byť prospešnou inšpiráciou pre ľudstvo. Mnohé z nižšie načrtnutých vlastností sa vyvinuli omnoho skôr, ako ich analogické podoby u ľudí, pomohli sa tak mravcom stať jednou z najúspešnejších existujúcich životných foriem.

Mravce, ktoré vznikli pred 100 miliónmi rokov zo samotárskych ôs sa vyznačujú podivuhodnými až neskutočnými vlastnosťami, ktoré stoja za zmienku a výskum.

Kým niektoré druhy pestujú huby – čo je celkom známy fakt, iné chovajú domáce zvieratá, ako sú vošky a rôzne húsenice. Podobne, ako v prípade húb, niekedy dochádza aj k tomu, že si kráľovná berie so sebou pri zakladaní nového hniezda aj domáce zvieratá. Niekoľko druhov mravcov je na svojom „dobytku“ celkom závislých a inak sa už ani nie sú schopní živiť. Chovaný hmyz rôzne ochraňujú, čistia, prenášajú z miesta na miesto, ba dokonca vešajú na správny druh rastlín, alebo priamo na ich stráviteľnú časť. Takéto chovanie vykazuje už len jeden druh na našej planéte, *Homo sapiens*.



Obr. 4 Mravec a jeho domáce zvieratá

Zvláštnou črtou niektorých jedincov je, že sa obetavo starajú nielen o svoje potomstvo, ale spolu so svojimi i o vajíčka a larvy vošiek listových. Regulujú aj populáciu červov, zabíjajú ich a pojedajú v prípade, že je červov viac, ako ich potrebujú na prísun medoviny. Túto ich vlastnosť využívajú sčasti i motýle, ktoré svoje vajíčka kladú v blízkosti mravcov, aby ich tak ochránili pred červami.

Ďalšou osobitosťou je aj ich starostlivosť o uskladnené semená, ktorým často obhrýzajú klíčky, aby tak predĺžili ich trvanlivosť.

Je zaujímavé, že pri niektorých druhoch mravcov sa kráľovná po oplodnení vracia späť do rodného hniezda a tak sa teritórium mraveniska dedí z generácie na generáciu.

Sledovaním mravcov faraónov sa zistilo, že ich novonarodené kráľovné sa pária so svojimi príbuznými z vlastného hniezda, čím sa stáva kolónia takmer nesmrteľná, avšak ustavične ohrozená inbreeding-om. Ak sa v tomto prípade kolónia rozdelí, zachováva sa medzi pôvodným a dcérinými mraveniskami priateľský vzťah, sú spolu prepojené vyznačenými cestami a navzájom si vymieňajú robotnice. Na ostrove Hokaido v Japonsku bola objavená taká super kolónia čiernych mravcov *Formica yessensis*, ktorá obsahovala približne 45 tisíc prepojených mravenísk, kde žilo okolo 300 miliónov jedincov.

Nevšedný spôsob preniknutia do cudzieho mraveniska si našli európske lúpežné mravce *Solenopsis fugas*, ktoré sa na protivníkov systém chodieb napájajú pomocou vyhĺbených tunelov.

Severoamerické púštne mravce z rodu *Canomyrma* hádžu počas svojich vojen kamienky do zvislých šácht nepriateľského hniezda. Zaraďujú sa tak do skupiny tých živočíchov, ktorí nejakým spôsobom využívajú nástroje.

Medzi kolóniami medonosných mravcov sa vyskytujú rituálne súboje, pri ktorých do seba navzájom narážajú súperiace skupiny a búšia sa tykadlami. Pritom si však nijak neubližujú. Víťaziace družstvo obsadí územie, o ktoré bojovali, no

akonáhle je súper schopný postaviť asi desaťkrát početnejšiu armádu, bez ohľadov a rituálneho súboja dobije nepriateľské mravenisko. Zabije kráľovnú a kohokoľvek, kto sa mu postaví na odpor, ukradne kukly aj mladšie robotnice.

Pri niektorých druhoch mravcov zakladá hniezdo spolu niekoľko kráľovien. Akonáhle je však prekonané prvé kritické obdobie (zaistenie územia a potravy, dostatok robotníč), rozpúta sa medzi nimi boj. Porazené kráľovné sú buď zabité, alebo vyhnané z mraveniska, no v prípade, že tam predsa zostanú, prestávajú klást' vlastné vajíčka. Sú známe i také prípady, keď si konkurujúce kráľovné navzájom ničia vajíčka.

Mravce zo strednej a južnej Ameriky dokážu dokonca i upravovať cesty. Štúdia, ktorej autormi sú Scott Powell a Nigel Franks z University of Bristol, uvádza, že mravce (*Eciton burchellii*) svoje pralesné komunikácie spravujú vlastnými telami. Lahnú si do výmoľov tak, aby bol povrch cesty rovnejší a ostatní (nájazdu sa v ich prípade môže zúčastniť až 200 000 jedincov) mohli za korisťou postupovať rýchlejšie. Ale i bez nejakého dramatického hromadného lovu sa takéto chovanie skrátka vyplatí – ostatní môžu po ceste efektívnejšie dopravovať potravu do mraveniska. Mravce, ktoré vyplňajú svojimi telami výmole, ku podivu nebývajú ušľapaní. Keď k večeru intenzita „dopravy“ poľaví, vylezú z dier a vrátia sa späť do mraveniska. Je zaujímavé, že dokážu rôzne kooperovať pri tom, ako zaplniť dieru väčšiu, než sú oni sami. Odhady autorov štúdie ukázali, že je naozaj výhodné, keď pár mravcov nezhromažďuje potravu, ale sa stará o komunikáciu a tým je finálne viacej zhromaždenej potravy.

Mravce dokážu vlastnými telami prekonať i vodný tok alebo vytvoriť akýsi živý rebrík medzi stromami, tam kde to inak nejde a na akcii sa zúčastňujú všetci. Vo vyššie spomínanom prípade je zaujímavé, že tá metóda slúži len k zvýšeniu efektivity. Je otázkou, či príslušné povolanie je pre jedinca trvalé, alebo či cestár môže svoju profesiu zmeniť podľa aktuálnych potrieb.

Sú to len náhodne vybrané aspekty zo života mravcov, mohli by sme v nich ďalej pokračovať, pretože ich život skrýva ešte množstvo záhad, ktoré nám môžu poslúžiť ako inšpirácia pri dodnes nevyriešených problémoch či optimalizácii tých vyriešených, alebo pri skúmaní sociálneho rozmeru živých spoločenstiev, zahŕňajúce i tú našu, ľudskú.



Obr. 5 Mravec ako súčasť cesty



Obr. 6 Mravčí most

Kapitola 3

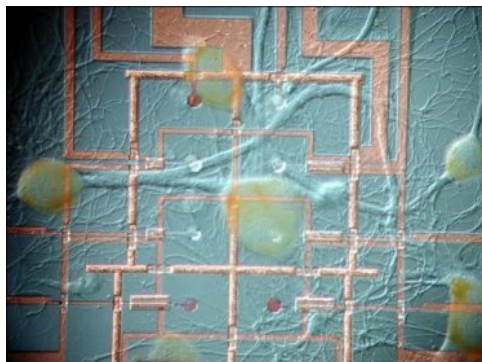
Umelý život

3.1 Inteligencia

Inteligencia – „ktorá sa meria testom inteligencie“

Počas skúmania evolúcie nervovej sústavy, spôsobov správania a poznania neustále narážame na otázku, čo je vlastne inteligencia. Odpoveď zďaleka nie je jednoduchá a závisí od toho, v akom vednom odbore sa angažuje ten, kto nám dáva odpoveď: psychológ, biológ, filozof alebo odborník na umelú inteligenciu. Klasický prístup sem zaraďuje predovšetkým vysokoúrovňové funkcie mysle spojené s vedomím, ako napríklad rozmyšľanie, uvažovanie, jazykové, logické a matematické schopnosti, ktoré charakterizujú predovšetkým alebo výlučne len ľudí. Na základe tohto sa i funkčný mechanizmus inteligentného chovania vysvetľuje v prvom rade pomocou príznakov ľudského uvažovania, manipuláciou symbolov, mentálnou reprezentáciou sveta. Tento prístup svedčí o nemalej odvahe, nakoľko za objekt svojej analýzy si vybral výstupy najzložitejšej „čiernej skrinky“ – ľudského mozgu, vyššiu úroveň fungovania mysle. Môžeme ho považovať za skúmanie nervovej sústavy z nadhľadu.

Z podobného základného postoja vychádza i výskum klasickej Umelej Inteligencie, ale v tomto prípade nie je prvoradým cieľom vysvetlenie fungovania, ale jeho reprodukcia [10]. Podmieňuje to predovšetkým inžiniersky prístup, ktorý rozkladá ľudský intelekt na samostatné komponenty, a pokúša sa reprodukovať jednotlivé kognitívne schopnosti ako na sebe nezávislé jednotky. K tomu nie je bezvýhradne potrebná znalosť činnosti ľudského mozgu, dokonca je možné ju úplne ignorovať. Filozofické pozadie pre tento uhol pohľadu ponúka funkcionalizmus, smer ktorý tvrdí, že funkcia mysle nezávisí od konkrétnej formy „wetware“-u, takýto istý výstup môžu, podobne počítačovým programom, ponúknuť i nosiče na úplne odlišnej báze, ich výsledky behov tiež nezávisia od toho, na akom stroji bežia. Filozof Hilary Putnam vyjadril tento fakt trochu výstredne: „Mohli by sme byť aj zo švajčiarskeho syra a nevadilo by to.“ Z podobného chovania, ktoré sa prejavuje na



Obr. 7 Wetware – Bio-elektronický obvod: živé bunky nervov (biologický wetware) integrované s mikroelektronickými obvodmi (elektronický hardware)

úrovni výstupov a z atitúdy výskumu z nadhľadu vyplýva otázka: „Je ľudská myseľ počítačový program?“ [11]. Táto otázka sa pre biológa môže zdať naivná, predsa sa stala základom rôznorodých filozofických debát [12, 13].

Inteligencia – adaptívne chovanie

Klasický prístup nie je schodná cesta pre tých, ktorí skúmajú inteligenciu z pohľadu evolúcie na základe jej pôvodu, počiatočných foriem prejavu a významu v prípade prežitia, alebo ho jednoducho považujú za príliš diskriminatívny. Veď nemáme žiaden dôvod predpokladať, že kognitívne schopnosti vyššej úrovne ľudí tvoria izolovanú skupinu javov a nemajú premisu u živočíchov s nižšou vyspelosťou [16, 14].

Akýkoľvek živočíšny druh si v záujme prežitia musí poradiť so všetkými výzvami, ktoré pred neho stavia jedno zložité, dynamické, nevyspytateľné udalosti obsahujúce prostredie. Čiže ak chceme skúmať evolúciu inteligentného chovania a nie len v podstate svojvoľne vybrané aspekty „vrcholu ľadovca“, je potrebné obohatiť inteligenciu o širšiu interpretáciu. Vychádzajúc z tejto potreby vznikol alternatívny pohľad, ktorý považuje za kritérium inteligencie adaptívne chovanie [14]. Toto samozrejme nie je novodobé poznanie, jeho pôvod siaha k počiatkom vedecky náročného výskumu živočíšneho chovania, nehovoriac o tom, že moderná etológia je veda sama o sebe evolučne motivovaná. Danú vlastnosť môžeme považovať za adaptívnu vtedy, ak prispeje k prežitiu daného živočicha. V prípade adaptívneho chovania je kladený dôraz na neustále vzájomné pôsobenie s prostredím, počas ktorého sa živočích nepretržite prispôbuje medziiným vnútorným a vonkajším podmienkam, ktoré sa sústavne menia práve kvôli stálej interakcii [14]. Tento prístup nepovažuje za kritérium inteligentného chovania tie črty, ktoré tvorili dôležité prvky klasického prístupu. Okrem adaptívneho charakteru chovania nestanovuje žiadne iné podmienky, ktoré by sa mali prejavíť vo vnútornej reprezentácii prostredia, v projekcii pomocou nejakých symbolov. Adaptívne chovanie prirodzene podmieňuje, aby dynamika vnútorných mechanizmov, ktoré toto umožňujú, v nejakej forme odpovedala dynamike prostredia (zhoda vytvorená evolúciou). Vonkajší pozorovateľ môže vidieť len to, že kým je chovanie adaptívne, existuje medzi daným živočíchom a prostredím určitá kompatibilita, a pokiaľ je táto prítomná, dovtedy spozorovateľné zmeny stavov v systéme vznikajú prostredníctvom dynamického vzájomného pôsobenia živočicha a prostredia. Ale akosť vnútorných mechanizmov zobrazenia, ktorá v určitej forme odráža štruktúru vonkajšieho prostredia, je možné považovať v tomto spôsobe pojednávania za irelevantnú [14], čo môže pre výskumníka znamenať rovnako výhodu ako nevýhodu.

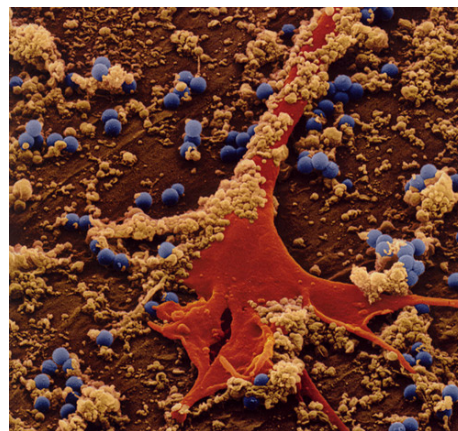
Opisovaný prístup je z viacerých pohľadov užitočný. Jednak poukazuje na adaptívny charakter inteligentného chovania, čím umožňuje jej pojednávanie v evolučnom rámci. Na druhej strane dáva priestor i pre jednotnú rozpravu prejavov živočíchov stojacich na rôznych stupniach vývoja. Popritom umožňuje i to, aby sme prejavy inteligencie nevzťahovali len na živočíšny svet a ľudí, ale aby sme ich rozšírili aj na všetky komplexné adaptívne systémy [17] od autokatalytických

chemických reakcií po supraindividuálne stupne organizácie [3], pomocou čoho sme schopní včleniť vývoj inteligencie a jej rôzne formy prejavu do obecnej teórie evolúcie [3, 17].

3.2 Čo je umelý život?

Jeden z novodobejších výsledkov symbiózy biológie a výpočtovej techniky je nepochybne smer, ktorý sa stal známym pod názvom umelý život (Artificial Life, A-Life). Tento, využitím výsledkov získaných počas výskumu genetických algoritmov a umelých neurónových sietí, ponúka nový nástroj na modelovanie biologických systémov a procesov. Nástroj, ktorý môže byť užitočný pre teoretickú biológiu, keďže vhodne aplikovaný je schopný zachytiť a skúmať komplexitu živých systémov. So systémom vo forme vhodného modelu sa omnoho ľahšie manipuluje, môže byť objektom takých precízne kontrolovaných pokusov, ktoré v prirodzených podmienkach často nie je možné uskutočniť [15]. No niektorí kritici poukazujú aj na fakt, že A-life je z metodologického hľadiska menej vypracovaný, z jeho výsledkov sa často usudzujú prehnané konklúzie. Je potrebné mať na zreteli aj to, že väčšiu časť výskumov vykonávajú informaticky (vyplýva to z charakteru metódy) a nie biologicky vzdelaní ľudia [16].

Situáciu ďalej komplikuje i to, že okruh ľudí zaoberajúcich sa touto tematikou nemá jednotnú motiváciu. Elliot Sober rozlišuje „silný“ a „slabý smer“ v rámci výskumu umelého života. Do prvej skupiny patria podľa neho tie programy, ktorých cieľom je modelovanie a skúmanie pravých živých systémov. Druhá obsahuje tie, ktoré ich autori považujú do určitej miery tiež za živé, ich primárnym cieľom nie je modelovanie už v prírode existujúcich biologických systémov, ale vytvorenie „alternatívnych životných foriem“ [7]. Vo vhodne vymedzenom priestore môže byť užitočným nástrojom aj druhý spomínaný smer, medziiným pri filozofickom skúmaní podstaty života, no otvára cestu aj k vytvoreniu „syntetickej biológie“ [7], ktorá môže plodne doplniť doteraz jediný, analytický prístup pri skúmaní živých systémov. Sú však i takí, ktorí by v ňom radi videli kybernetické eko-náboženstvo budúcnosti (cybernetic eco-religion of the future), preto je dôležité, aby sa možnosti odhalené prostredníctvom počítačov stretli s teoretickou pripravenosťou [16]. Podľa Millera má „slabý smer“ väčšiu šancu na to, aby sa po vykryštalizovaní výskumných metód stal osožnou a uznávanou časťou teoretickej biológie. V kriticky ladenom článku vymenoval niekoľko takých oblastí, kde by mohol byť A-life najúspešnejší. Medzi tieto patrí napr. otázka evolúcie správania sa, spoznávania a nervovej sústavy, ktoré sú priveľmi zložené na to, aby



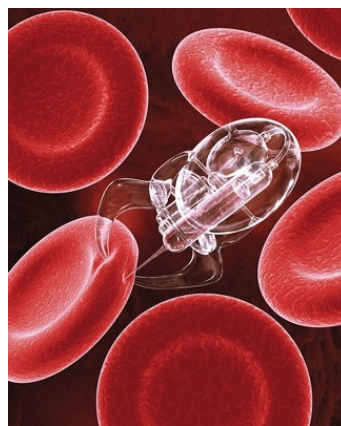
Obr. 8 Segment nervovej sústavy, neuróny spojené elektrochemickými signálmi

sme ich mohli úspešne skúmať tradičnými nástrojmi. Vieme len málo o vývoji behaviorálneho pozadia nervstva, nakoľko neuroetológia a porovnávacia psychológia má možnosť skúmať len súčasný stav nervovej sústavy a evolúcie kognitívnych procesov. V tejto otázke poskytujú aj fosílie len veľmi málo oporných bodov [17]. Systémy A-life sú schopné modelovať evolúciu nervstva a to v súvislosti s interakciou jedincov medzi sebou ako aj prostredím. To nám môže pomôcť lepšie pochopiť spoločný vývoj nervovej sústavy, chovania a inteligencie [16].

Rozdiely medzi umelým a prírodným sú na úrovni tvarových a funkčných primitív veľmi zreteľné, na úrovni súčinnosti primitív sú už omnoho menšie a nakoniec na úrovni nimi syntetizovaných procesov, javov a štruktúr sa rozdiely úplne strácajú [18].

Podľa Raya [7] existujú 3 základné formy syntetického prístupu:

- in hardware – robotika, nanotechnológia
- in software – počítačové programy, vykazujú isté črty „života“
- in wetware – replikujúce sa a vyvíjajúce sa makromolekuly, RNK svet



Obr. 9 Budúcnosť nanotechnológie podľa Coneyly Jaya

Táto práca sa v drvivej väčšine opiera o poznatky z druhej formy, ktorá sa ešte ďalej delí na simuláciu a na inštanciovanie procesov života.

Pri procesoch života simulovaných klasickým spôsobom sa vychádza zo sústavy diferenciálnych rovníc, ktoré vyjadrujú vzájomné vzťahy a chovanie jednotlivých zložiek (napr. génov, jedincov, druhov, atď.). Tento prístup simulácie sa zakladá na syntéze smerom zdola hore, kde sa najprv vytvorí populácia dátových štruktúr, pričom každá inštancia štruktúry predstavuje jednu entitu. Stav jednotlivých entít je obsiahnutý v premenných daných štruktúr. Následne stanovíme pravidlá, podľa ktorých sa riadi interakcia medzi entitami navzájom a entitami a prostredím. Počas simulácie sa populácia entít postupne mení vplyvom aplikácie pravidiel a na globálnej úrovni je možné pozorovať výsledné chovanie, ktoré vzniká na základe lokálnych interakcií.

Uvedené dátové štruktúry reprezentujú v prípade simulácie nejakú skutočnú, živú entitu, ako sú napr. mravce. V prípade inštanciovaných prejavov života nereprezentujú nič iného okrem seba samých. Takéto štruktúry potom pokladáme za „živé“ a sledujeme, ako sa ich lokálne chovanie a interakcia prejaví na globálnej úrovni. Podľa tejto úvahy je možné tvrdiť, že prirodzený život na Zemi a niektoré umelé formy života sú rôznymi inštanciami života ako takého [18].

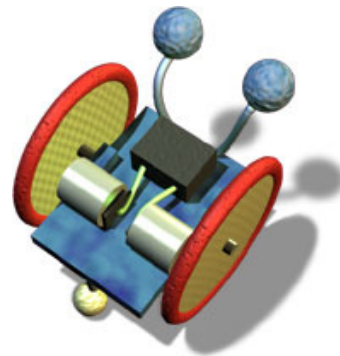
Základné tézy umelého života podľa Júliusa Csontóa [18]:

- Informácia, a nie materiálna forma slúžiaca k jej uchovávaniu a spracovaniu, je podstatou života.

- Život podmieňuje určitý stupeň zložitosti – štruktúra získava po jej dosiahnutí schopnosť rozmnožovať sa, a to nie len vytváraním svojich identických kópií, ale i tak, že potomkovia, ktorí vzniknú sa stanú zložitejšími.
- V živých organizmoch má informácia dvojitú podobu: genotyp a fenotyp. Genotyp – neinterpretovaná informácia slúži k rozmnožovaniu, je odovzdávaná potomkom. Fenotyp – interpretovaná informácia slúži ako základ pre vytvorenie štruktúry nového jedinca.
- Samoreprodukcia, mutácia a selekcia je v prípade všetkých foriem života prostriedkom vývoja smerom k zložitejším, dokonalejším štruktúram – evolúcii. Evolúcia môže byť v prípade umelého života nie len darwinovská, ale napr. aj lamarckistická, poprípade môže vykazovať tzv. Baldwinov efekt (poznatie, ktoré jedinec získa učením sa síce priamo nededí, ale zvyšuje jeho šancu na rozmnožovanie, čím sa v ďalších generáciách zvýši výskyt génov zodpovedných za lepšiu schopnosť učenia).
- Syntetický proces prebieha „in vivo“ i v pokusoch „in vitro“ a „in silico“ (v počítači) zásadne smerom zdola hore: od elementárnych primitív, ktoré sú riadené jednoduchými pravidlami až k zložitým štruktúram vykazujúcim komplexné chovanie. Táto metóda sa diametrálne odlišuje od procesu zhora dolu, ktorý je charakteristický pre väčšinu postupov umelej inteligencie.
- Pre dynamiku lokálnych primitív je charakteristická vysoká miera paralelizmu.
- Vzájomné lokálne interakcie primitív produkujú na globálnej úrovni úplne nové fenomény. Tento jav sa označuje ako emergencia, všetko pritom prebieha bez akéhokoľvek centrálného riadenia.
- Nelineárne chovanie elementárnych primitív, čiže neplatnosť princípu superpozície (ktorého predpokladom je, že chovanie celku je možné odvodiť zložením jednotlivých chovaní primitív) je jednou z podmienok emergencie.

Komunikácia a interakcia s inými jedincami a prostredím je základnou vlastnosťou každej vyššej životnej formy. Z hľadiska ľudského poznania je nesmierne dôležitý sociálny faktor. Tieto fakty motivujú vytvorenie takých modelov, kde dostáva úlohu interakcia, kde je každý jedinec aktívnou časťou nejakého systému podobne, ako je tomu pri živých systémoch.

Jeden z prvých pokusov sa spája s menom biológa Valentina Braitenberga, v ktorom vystupujú jednoduché „vozidlá“, roboti. Tieto sú schopné vnímať vonkajšie podnety (zdroj svetla, pohyb, prítomnosť iných jedincov) a dopredu naprogramovane odpovedať („agresívne“, „zvedavé“, „milé“, atď. pohybové vzory). Braitenbergov pokus dobre ilustruje jednu zo základných vlastností A-life: systém skladajúci sa z jednoduchých a interaktívnych častí môže vykazovať komplexné a emergentné chovanie, je viac, ako len súčet jeho komponentov.

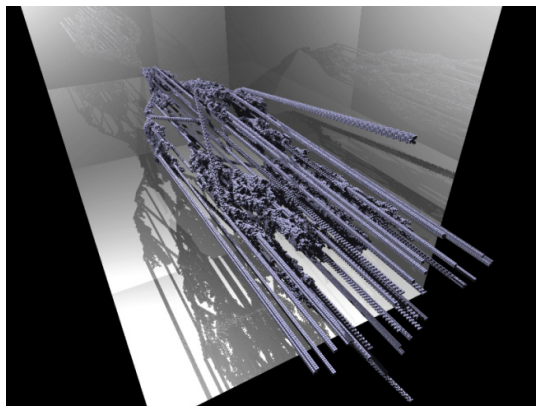


Obr. 10 Príklad vozidla Valentina Braitenberga

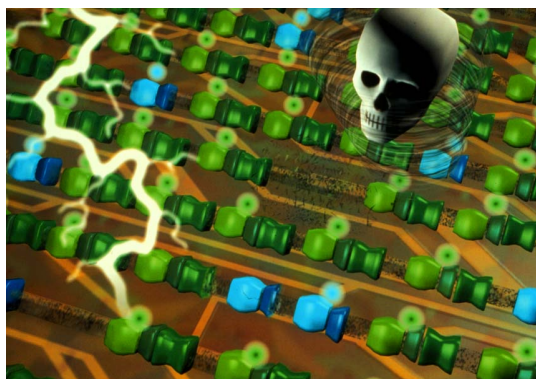
3.3 Evolúcia v počítači – „in silico“

Javiskom experimentov umelého života sú popri skúmaní robotov i počítačmi simulované umelé svety a v nich pôsobiace umelé bytosti, ktoré sledujú také ciele, ako je získavanie potravy, rozmnožovanie sa a najviac inteligencie podmieňajúce prežitie. Takéto simulátory umelého života sú napr. Tierra, Avida, Lee, Swarm, AI systém, či klasická Conwayova hra života, celulárne automaty alebo komunita umelých mravcov. V takýchto a tomu podobných systémoch dostáva rolu evolúcia – chovanie umelých bytostí určuje náhodne resp. dedične daná „genetická sada“. V boji o potravu, párenie sa, životný priestor atď. zmiznú nevýhodné vlastnosti, počas opakovania simulácie z generácie na generáciu sa vyvinú najvhodnejšie varianty (adaptácia).

Medziiným aj tieto myšlienky viedli ku genetickým algoritmom (GA), ktoré sú spájané s menom Johna Hollanda. Podstata GA, ktoré považujú v umelej inteligencii resp. informatike za efektívnu programovaciu techniku, spočíva v riešení úloh založených na jednoduchých selektívnych mechanizmoch. Nejaká dobre definovaná fitness funkcia selektuje medzi „horšími“ a „lepšími“ riešeniami, dochádza k páreniu jednotlivcov medzi sebou, ich génový základ sa mutáciou a krížením (crossover) mení, pokiaľ sa nenájde optimálne riešenie alebo sa nesplní nejaké iné kritérium (napr. až príliš homogénna sada génov populácie pre ďalší vývoj).



Obr. 12 Game of life – s pridaným časovým rozmerom



Obr. 11 Snímok z Tierra Video: digitálne prostredie – samoreplikujúce počítačové programy (farebné geometrické objekty) okupujú pamäť RAM (oranžové pozadie). Mutácia náhodne mení kód (blesk), smrť (lebká) odstraňuje staré a chybné programy.



Obr. 13 Genetická operácia – mutácia

Agenti, podobne ako umelé bytosti, sú naprogramovaným chovaním disponujúce aktívne jednotky, ktoré sa dostávajú do interakcie so sebou a svojim prostredím. Môžu byť schopní učiť sa, adaptovať sa (ako v darwinovskom, tak aj v skinnerskom či popperskom poňatí). Interakciu viacerých agentov modelujúce štruktúry sú multiagentné systémy, ktorých globálne, emergentné vlastnosti môžu byť pre nás zaujímavé. Dnes nám počítače umožňujú simuláciu a výskum takých systémov, ktoré obsahujú státisíce až milióny agentov. Agenti a multiagentné systémy sa dnes už využívajú aj vo výskumných oblastiach ekonómie, ekológie, populačnej dynamiky alebo rôznych aplikácií informatiky.

Syntetická evolúcia

Informácie získané o životných javoch veľkou mierou obmedzuje fakt, že máme možnosť skúmať len jednu realizáciu – zemský život. Komparatívne výskumy by nám mohli pomôcť lepšie pochopiť všeobecné zákonitosti.

Iné možné životné formy:

- mimozemský život – keď existuje, tak je pravdepodobne podobný zemskému životu, založenom na uhlíku.
- umelý život – v úplne odlišnom prostredí vytvorená životná forma by bola ideálna k porovnávaciemu výskumu. Je to syntetický, konštruktívny prístup oproti klasickému, prírodovedeckému redukcionistickému, analytickému prístupu.

Niekoľko z výhod evolúcie a života v počítači:

- jednoduché získavanie dát
- evolučný proces je možné meniť aj počas chodu
- experimenty je možné presne zopakovať
- proces je rýchly (je otázne, či sa úmerne zníži rýchlosť, keď digitálne systémy dosiahnu zložitosť organického života)

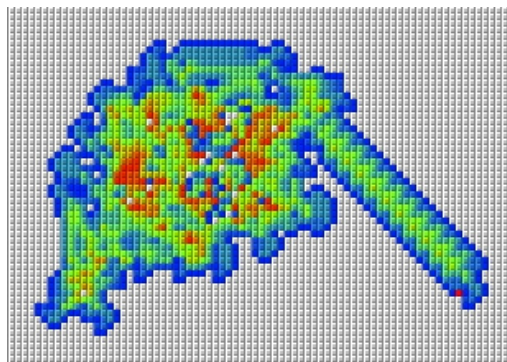
Kapitola 4

Prístupy riešenia problematiky

4.1 Existujúce aplikácie, možnosti riešenia

Vants – Virtual Ants

Autorom tejto simulácie je Christopher Langton [19], jedna z ústredných postáv celého hnutia A-life. Je to simulácia chovania virtuálnych mravcov – Vantov (z Virtual Ant). Vant je „organizmus“ tvaru písmena V. Pohybuje sa v smere vrcholu V vo svete pozostávajúceho zo štvorcových políček usporiadaných do pravidelnej mriežky. Keď sa dostane na prázdne políčko mriežky, pokračuje



Obr. 14 Vants – Virtual Ants

v pôvodnom smere. V prípade, ak bolo toto políčko modré, Vant zabočí doprava a farba políčka sa zmení na žltú. Keď postúpi na žlté políčko, zabočí doľava a zmení farbu na modrú. Týmto spôsobom zanecháva Vant za sebou „feromónovú“ stopu. Navyše je schopný v takomto prostredí identifikovať, či bolo políčko prejdené párny alebo nepárny krát. Aj v prípade jediného Vanta tak vzniká pomerne komplexné správanie, pretože neustále naráža v prostredí na výsledok svojho minulého správania. V prípade spolupôsobenia viacerých virtuálnych mravcov vznikajú ešte komplexnejšie vzory. Prekvapivo, už i takto jednoduché pravidlá produkujú výsledné kooperatívne chovanie veľmi podobné chovaniu sociálneho hmyzu, živých mravcov. Spočiatku váhavo kľúčujú, postupne začínajú interagovať, až napokon spoločne vytvoria špirálovitú stopu.

Týmto sa Langton úplne presvedčil o efektívnej aplikácii modelu celulárneho automatu, o kolektívnom systéme bez riadenia centrálnej inteligencie skladajúcej sa z drobných osobitých činností, ako aj o zdola hore prebiehajúcom (bottom-up), emergentné vzory chovania produkujúcom prístupe. Dospel k takému výsledku, ktorý dopredu nenaprogramoval.

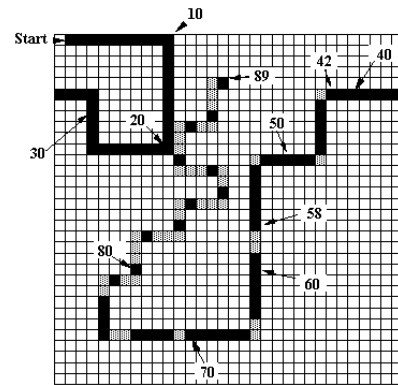
Genesys system

Je ďalším simulátorom tejto kategórie. Simuluje vlastnosť mravcov evolučne sa prispôbovať zadanej trase [20]. Svet systému je štrukturálne orientovaná 2–dimenzionálna diskretná mriežka. Mravec je simulovaný v izolácii ako sa pokúša

nasledovať stále redšiu stopu potravy. Nie sú modelované žiadne fyzikálne vlastnosti.

Hlavným cieľom systému je porovnať dva typy kontrolných mechanizmov slúžiace k riadeniu chovania mravca počas nasledovania stopy: konečný automat (disponujúci 16 stavmi) a rekurentná umelá neurónová sieť. Oba typy sú explicitne kódované v genotypе mravca. Konečný automat je reprezentovaný ako tabuľka (binárne kódy) stavových prechodov, pričom neurónová sieť je kódovaná ako váha, prah, a počiatočná aktivácia pre každú jednotku (z jedenástich).

Systém Genesys ešte nebral do úvahy jeden veľmi dôležitý moment, a to sociálne kooperatívne chovanie.



Obr. 15 John Muirova stopa v 32x32 mriežke s vyznačenými hodnotami bodov za dosiahnutie daných políček

Manta

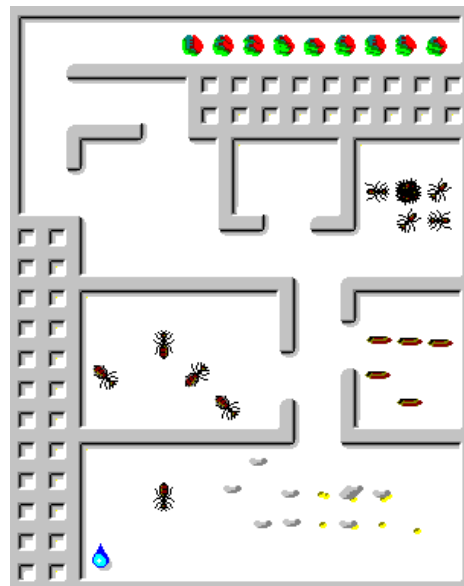
Modeling an ANTnest Activity – je jedným z najtypickejších simulátorov tejto oblasti, ktorý simuluje správanie sa kolónie mravcov [21]. V týchto kolóniách sa kladie dôraz na vykonávanie spoločných úloh, ktoré presahujú rámec jednotlivých jedincov. Túto spoluprácu realizujú bez toho, aby boli zapojení do špecifickej hierarchickej architektúry, bez centrálného riadenia a s veľkým vnútorným šumom. Práve kvôli takejto organizácii sú základom simulátora reaktívni agenti. Každý agent je autonómna bytosť, ktorá je schopná rozosielať a prijímať správy, nedokáže sa učiť alebo „myslieť“, jednoducho len reaguje na správu stereotypným chovaním.

Manta obsahuje tri druhy agentov:

- asistenti: kráľovná, robotnice, trúdi
- asistovaní: larvy v rôznych štádiách vývoja
- fyzikálne prostredie: svetlo, vlhkosť, mŕtvi jedinci

Systém je objektovo orientovaný, charakteristiky agentov tvoria hierarchickú štruktúru, pri ktorej sa uplatňuje dedičnosť. Agenti disponujú:

- spoločným abstraktným spôsobom chovania,
- grafickou reprezentáciou,
- priestorovým umiestnením,
- schopnosťou dospievať.



Obr. 16 Prostredie systému MANTA

Simulácia väčšinou začína samostatnou kráľovnou, ktorá je situovaná do prostredia so zadanými fyzikálnymi parametrami (vlhkosť, svetlo, atď.). Obrázok č. 16 znázorňuje príklad prostredia simulácie.

Prakticky sa tento systém využil pri skúmaní sociogenézy osíh kolónií. Sociogenéza je proces, pri ktorom jeden jedinec – gravidná kráľovná – vytvorí celú kolóniu. Je to veľmi jemný proces z bioenergetického hľadiska. Spočiatku je pre kráľovnú jediným prameňom energie degenerácia jej svalov na krídlach. Potom, keď sa z vajíčok vyvinú mravce, začnú zbierať potravu a kŕmiť kráľovnú. Kráľovná v tomto štádiu čiastočne optimalizuje znášanie potravy do mraveniska, kladenie vajíčok, prenášanie vajíčok a lariev.

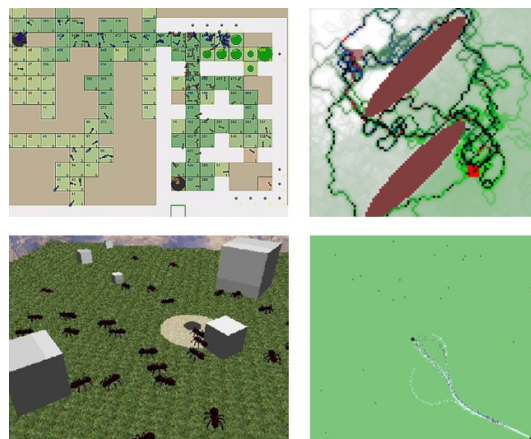
Behom týchto pokusov sa dospelo k názoru, že je možné dosiahnuť kvalitu prirodzenej sociogenézy ako v skutočnej populácii živých jedincov. Spoločenstvo mravcov môže byť programovo reprezentované hierarchickou štruktúrou agentov. Tento konkrétny prípad naznačuje silu a hranice simulácie skutočného života.

AntFarm

Jedným z najprepracovanejších simulátorov tematiky je AntFarm [22], ktorý je implementovaný na mohutnom paralelnom prostriedku – Connection Machine [5]. Simuluje súčasne 16 348 kolónií po 128 mravcoch, genóm každého tvorí 25 590 bitov (vo vyššie popisovanom systéme Genesys ich bolo iba 450). Mravenisko a jeho okolie tvorí mriežka 16 x 16 buniek. Genóm každého jedinca z jednej kolónie je rovnaký. Jednotliví jedinci sú riadený rekurentnou neurónovou sieťou, ktorá má na vstupe informácie zo senzorov o potrave a o feromónovej stope v okolí 3 x 3, o tom, či jedinec nesie potravu, o relatívnej polohe voči mravenisku. Mravec disponuje pamäťou veľkosti 21 bitov. Spočítané výstupy siete ovládajú motory, ktoré slúžia k pohybu mravca, manipulácii s potravou a k vylučovaniu feromónu.

Kolónia sa vyvíja ako jeden veľký superorganizmus pomocou genetického algoritmu, ohodnocovaného funkciou fitness, ktorá je definovaná predovšetkým schopnosťou danej kolónie zhromažďovať potravu, nárokmi na pohyb či na vylučovanie feromónov.

Okrem hore uvedených je možné spomenúť ešte niekoľko ďalších systémov tejto oblasti: demonštračná aplikácia k práci Simulation of Artificial Ant Colonies H. Ahvaziho, Ants Box Simulator P. E. Merlotiho, Ant Farm Simulator od R. A. Maturanu, MASON – multiagentný simulátor implementovaný na George Mason University či demonštračná simulácia práce Food-gathering in Ant Colonies od S. Tangemana.



Obr. 17 Ukážky aplikácií – v poradí: Ant Farm, MASON, Food-gathering in Ant Colonies, Ants Box

4.2 Vlastné vybrané postupy riešenia

Po preskúmaní problematiky a po zrelej úvahe som sa rozhodol vybrať pre porovnanie dva najslubnejšie prístupy riadenia jedincov, a to: konečný automat (jeho jednoduchosť dobre odráža jednoduchosť samotných mravcov) a evolvujúca sa umelá neurónová sieť („nechať riešenie na prírodu“, komplexné adaptívne systémy majú tiež evolučný charakter). Tieto dva zvolené metódy riadenia boli implementované v rámci simulačného nástroja – aplikácie (SocInCo), ktorého riadenie a spôsob realizácie sa podrobne pojednáva v prílohách práce (programátorská a užívateľská dokumentácia).

Konečný automat

Konečný automat (FSM z angl. finite state machine) je teoretický výpočtový model používaný v informatike pre štúdium vyčísliteľnosti a všeobecne formálnych jazykov. Popisuje veľmi jednoduchý počítač, ktorý môže byť v jednom z niekoľkých stavov a medzi nimi prechádza na základe symbolov, ktoré číta zo vstupu. Množina stavov je konečná (odtiaľ má názov). Konečný automat nemá okrem informácie o aktuálnom stave žiadnu ďalšiu pamäť. Je to veľmi jednoduchý výpočtový model, dokáže rozpoznávať len regulárne jazyky.

Formálne je konečný automat definovaný ako usporiadaná päťica $(S, \Sigma, \sigma, s, A)$, kde:

- S je konečná množina stavov
- Σ je konečná množina vstupných symbolov, nazývaná abeceda
- σ je tzv. prechodová funkcia, popisujúca pravidlá prechodov medzi stavmi
Môže mať buď podobu $S \times \Sigma \rightarrow S$ (deterministický automat), alebo $S \times \{\Sigma \cup \varepsilon\} \rightarrow P(S)$ (nedeterministický automat)
- s počiatočný stav, $s \in S$.
- A je množina prijímacích stavov, $A \subseteq S$

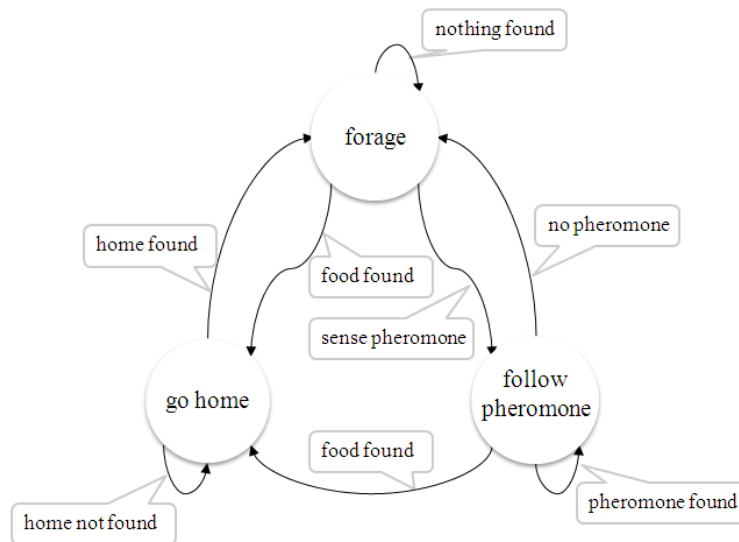
Na počiatku sa automat nachádza v definovanom počiatočnom stave. V každom kroku prečíta jeden symbol zo vstupu a prejde do stavu daného hodnotou odpovedajúcou aktuálnemu stavu a prečítanému symbolu v prechodovej tabuľke. Potom pokračuje čítaním ďalšieho symbolu zo vstupu, a následne ďalším prechodom podľa prechodovej tabuľky.

Podľa toho, či automat skončí po prečítaní vstupu v stave, ktorý patrí do množiny prijímacích stavov, platí, že automat buď daný vstup prijal alebo neprijal. Množina všetkých reťazcov, ktoré daný automat prijme, tvorí regulárne jazyky. Rozšírený konečný automat neprijíma v tomto experimentálnom kontexte regulárne jazyky, skôr špecifikuje akcie ako odpovede na vstupy agentov.

Nakoľko je činnosť mravcov riadená jednoduchou sadou pravidiel, implementácia ich chovania konečným automatom sa priam núka. Komplexný problém sa rozčlení na niekoľko podproblémov.

Namiesto relatívne neprehľadného popisu konečného automatu priamo tabuľkou sa obvykle používa (i v tomto prípade, obr. 18) grafické znázornenie, orientovaný graf, na ktorom kolieska znázorňujú jednotlivé stavy a šípky (s pridruženým

vstupným symbolom, resp. vstupom agentov) medzi týmito kolieskami popisujú jednotlivé prechody.



Obr. 18 Schéma konečného automatu

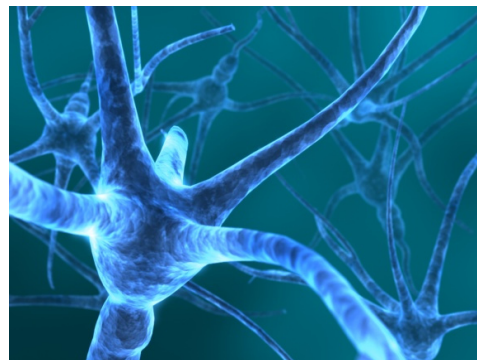
Mravce sa narodia do stavu hľadania potravy (počiatočný stav) a začnú svoju „náhodnú“ púť za potravou. Keď sa dostanú do kolízie s hranicami prostredia, otočia sa a pokračujú v hľadaní. Ak narazia na potravu, vezmú ju a prejdú do stavu späť s potravou domov. Po zložení potravy v mravenisku sa vrátia do stavu hľadania potravy. Keď mravec zdetekuje feromón, prepne sa do stavu nasledovania feromónu. (časť z vývojovej dokumentácie)

Umelá neurónová sieť v kombinácii s genetickými algoritmi

Spojenie umelých neurónových sietí a genetických algoritmov môže byť z hľadiska evolúcie vhodným nástrojom pre modelovanie vzťahu nervovej sústavy a adaptívneho chovania. V centre našej pozornosti stojí vývin a ďalší rozvoj najstaršieho vývojového stupňa nervstva, čiže najzákladnejšie zásady organizovania sa a konštrukčné pravidlá, ktoré uložením v genóme postačujú k vytvoreniu funkčnej nervovej sústavy bez toho, aby sme museli detailne určovať výstavbu nervstva.

Umelé neurónové siete sa môžu zdať obzvlášť vhodným nástrojom na to, aby sme pomocou nich modelovali vznik a činnosť jednoduchých sieťovej nervovej sústavy.

Som si vedomý toho, že úplné a detailné vypracovanie takéhoto modelu by značne presahovalo rámec tejto práce, jej cieľom je preto len polozenie základov, vytvorenie takej verzie, ktorej výsledky je možné vyhodnotiť, vyvodiť dôsledky, aby tak mohla tvoriť podklad budúcich prác.



Obr. 19 Neuróny

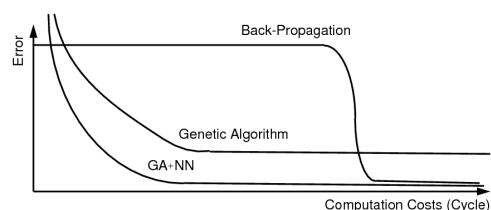
Motivácia kombinácie

Vidiac popularitu a úspechy darwinovskej teórie prirodzenej selekcie nie je prekvapujúce, že ovplyvnila spôsob myslenia výskumníkov aj v ostatných oblastiach vedy. Veď v podstate elegantne zodpovedá otázku, ako sa dajú spomedzi naskytujúcich sa riešení s ohľadom na daný problém vybrať tie, ktoré je možné za daných podmienok považovať za najviac efektívne.

Metóda prirodzenej selekcie môže byť vhodná aj pri riešení matematických či programátorských úloh. K tomuto záveru došiel v 60-tych rokoch i John Holland, otec genetických algoritmov. Ich rysy sú nasledovné: je daná úloha, pre ktorú začiatkom každého cyklu vygenerujeme veľké množstvo potencionálnych riešení tak, že rôzne riešenia danej úlohy navzájom skombinujeme, resp. na prvkoch riešenia vykonáme náhodné zmeny. Takto vytvorené „návrhy riešenia“ testujeme, následne z určitého hľadiska najlepšie riešenia podrobíme ďalším cyklom. Zaujímavosťou tejto metódy je, že aj vtedy je počítač schopný úspešne riešiť úlohu, ak pôvodne neboli k dispozícii pravidlá, ktoré toto umožňovali. Disponuje sadou, ktorá obsahuje prvky použiteľných pravidiel, čiže má schému riešenia, a ich kombináciou, resp. mutáciou vzniknutými novými prvkami môže byť schopný dospieť k správne riešeniu. Podobne ako aj u biologickej evolúcie, aj tu má kľúčovú úlohu mutácia, pretože umožňuje, aby algoritmus neuviazol na nižších vrcholoch fitness-krajinky. Táto krajinka vzniká priradením fitness-hodnôt genotypom, ktoré reprezentujú „vhodnosť“ nimi zastúpeného riešenia. Jej reliéfové pomery označujú hodnoty fitness, čiže genotypy najlepších riešení sa nachádzajú na najvyšších „vrcholoch“ a práve tie predstavujú cieľ bádania.

Objavenie genetických algoritmov dalo efektívny nástroj do ruky tým, ktorý sa zaoberali riešením problémov, kde je potrebné vybrať z veľkého počtu možných riešení za daných podmienok tie najlepšie [22]. Presne takúto úlohu predstavuje plánovanie a nastavenie počiatočných hodnôt váh (v aplikácii je týmto spôsobom implementované učenie) pri použití umelých neurónových sietí pre rôzne účely [23]. Čiže výskumníkov umelej inteligencie riadia pri spájaní týchto dvoch typov modelov praktické hľadiská a nie požiadavka dokonalej biologickej vernosti [24]. Naproti tomu odborníci, ktorí považovali neurónové siete za model nervovej sústavy, v nich videli evidentnú možnosť s ich pomocou skúmať svoje modely z evolučného hľadiska. V oboch výskumných smeroch je však najprv potrebné vyriešiť problém kódovania neurónových sietí v genómoch a tiež určiť hodnotu fitness jednotlivých genotypov. Tieto je možné podľa cieľu modelovania zrealizovať rôznymi spôsobmi, čo patrí do tematického okruhu evolučného plánovania neurónových sietí.

Výhodnosť kombinácie potvrdzujú i experimenty, ktoré vykonal začiatkom 90-tych rokov Hiroaki Kitano. Tie ukázali, že kombinácia (GANN) je výhodnejšia ako samotné neurónové siete (NN) či genetické algoritmy (GA). Ja si však myslím, že by bolo nebezpečné vyvodzovať unáhlené



Obr. 20 Výsledky pokusov H. Kitana

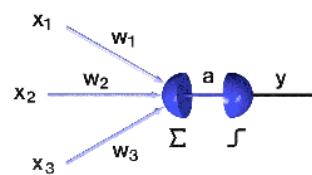
závery. Je pravdepodobné, že vo väčšine prípadov to nedopadne ani zďaleka tak jednoznačne, tak výhodne pre GANN. Hlavne v prípadoch, keď sa proces dostane do lokálneho minima, z ktorého sa len ťažko vráti späť k zlepšovaniu.

Ďalším argumentom pri voľbe tohto postupu je, že neurónové siete zastupujúce jedincov tohto typu nie je možné učiť klasickými metódami. Chýbajú nám správne výstupy siete, nie je možné určiť odchýlku od správneho stavu.

Formálna definícia NN

Neurónová sieť je masívne paralelný výpočtový systém, ktorý má schopnosť uchovávaní informácií a umožňuje ich ďalšie spracovanie, pričom v zbieraní poznatkov v procese učenia a uchovávaní týchto poznatkov s využitím medzineurónových spojení napodobňuje ľudský mozog.

Je výpočtový model pozostávajúci z množstva pospájaných prvkov, známych ako neuróny. Neurón je spracovateľská jednotka prijímajúca vstupy z vonkajšieho prostredia siete alebo z iných neurónov, lokálne ich transformuje a poskytuje výstupný signál, ktorý prechádza na ďalšie neuróny alebo mimo sieť. Každý zo vstupov neurónu je modifikovaný hodnotou súvisiacou so spojením. Táto hodnota sa vzťahuje na silu spojenia alebo váhu, reprezentuje koľko dôležitosti neurón pripisuje tomuto vstupu. Operácia, ktorou neurón transformuje svoje vstupy na výstup je spravidla veľmi jednoduchá. Zložitá činnosť neurónovej siete spočíva v spojení mnohých takýchto jednoduchých elementov do celku – v prevažnej väčšine sietí sú neuróny usporiadané do vrstiev.



Obr. 21 Umelý neurón

Typická neurónová sieť je schopná reprezentovať viaceré funkcie, ale objavenie tej najlepšej siete potrebnej na vyriešenie špecifického problému je doposiaľ nedoriešený problém. Keby sme poznali presnú metódu riešenia, tak by sme vedeli naprogramovať štruktúru siete explicitne. Avšak, ak je tento problém veľmi komplexný alebo nemá žiadne doteraz poznané riešenie, nemôžeme vedieť akú štruktúru máme sieť dať. Za týmto účelom, väčšina modelov neurónových sietí zahŕňa pravidlo učenia, ktoré môže zmeniť štruktúru siete v priebehu učenia a prísť tak na dobré koncové riešenie. Toto učiace pravidlo môže predstavovať genetický algoritmus, ktorý môže byť použitý na optimalizovanie neurónovej siete rozličnými spôsobmi: evolúcia váh (použitý prístup), evolúcia štruktúry, evolúcia pravidiel učenia.

V prípade evolúcie váh populáciu GA tvoria neurónové siete s rovnakou základnou topológiou, ale s rozličnými hodnotami váh. Mutácia a kríženie teda ovplyvňujú len váhy, ktoré sú v NN ako jedince. Kľúčovou otázkou v takomto systéme je kódovanie sietí, ktoré sa pojednáva nižšie. Použitie GA namiesto gradientných metód tréningu neurónových sietí vedie k rýchlejšej a lepšej konvergencii. Predsa len, keďže GA sú dobré na globálne prehľadávanie ich použitie je neefektívne pri lokálnom prehľadávaní. V takom prípade je vhodné využiť hybridný prístup kombinujúci GA s gradientnými metódami.

Pseudo-kód evolúcie neurónových sietí

```

g = 0          /* inicializacia pocitadla generacii */
P = Initialize population ()
              /* pociatocná generacia jedincov - chromozomy
              mravcov reprezentovane vahami ich sietí */
Repeat until end condition
  T = Training (P)          /* nasadenie v "praxi" */
  E = Evaluation (T)        /* ohodnotenie chromozomov */
  R = Ranking (E)           /* poradie na zaklade fitnessu */
  S = Selection (R)         /* ruletova selekcia */
  C = Crossover (S)         /* dvojbodove krizenie - moze byt
                          NOP: kvoli elitizmu alebo
                          vyhradnej mutacii */

  M = Mutation (C)          /* mutacia s danou pravdepodobnostou
                          - moze byt NOP: kvoli elitizmu
                          alebo vyhradnemu krizeniu */

  P = Update population (M)

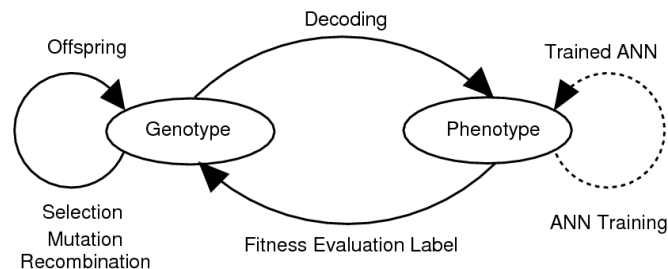
                          /* M sa skopiruje do P (P = M) */
g = g + 1          /* zvysi citac generacii */

```

Detaily a schéma algoritmu sa pojednávajú tiež v dokumentácii aplikácie.

„Životný cyklus“ neurónových sietí – sklbený s GA

Počas evolučného modelovania neurónových sietí má aktivita dve scény. Jednou je úroveň genotypu, kde sa pomocou genetických operátorov deje projekcia genotypu do fenotypu, čiže dekódovaním informácií uložených v genóme vznikne neurónová sieť – proces „dozrievania“ (maturation). Druhou je úroveň fenotypu, kde už funguje vytvorená neurónová sieť. Vyhodnotením výsledkov tohto fungovania sa stanoví fitness, ktorý určí ďalší osud daného genotypu (obr. 22) [23]. Procesy prebiehajúce na týchto dvoch úrovniach spolu tvoria vývoj neurónových sietí (development).



Obr. 22 Schéma evolúcie neurónových sietí

V prípade živočíchov, nie je možné tieto dve úrovne rozlíšiť, no pri modeloch je spozorovateľná výrazná separácia: najprv dôjde k vytvoreniu fenotypu a len potom sa uvedie do chodu. Toto sa pokúšajú niektoré modely zmeniť a spraviť dozretie pravou ontogenézou tým, že predĺžia formovanie siete v čase, a takto dosiahnu úplne vyvinutú formu cez sériu sekvenčne nasledujúcich rôznych stavov vývoja. Počas tohto vývinu je sieť už v prevádzke, teda vytvorenie konečného fenotypu ovplyvňujú i vonkajšie pôsobenia. Dôsledkom toho dostanú vysoké hodnoty fitnessu iba tie siete, ktoré už i v ranných štádiách svojho vývoja dávajú adaptívne odpovede na podnety prostredia, alebo tie, ktoré prejdú takým efektívnym procesom učenia, ktorého výsledky sa vo veľkej miere pričinia o rast ich fitnessu v neskorších etapách ich života.

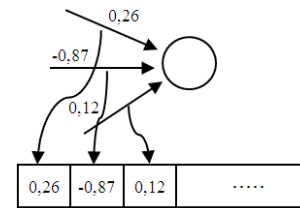
Problémy kódovania sietí

Podstatná otázka z hľadiska fungovania modelu je, akým spôsobom je kódovaná neurónová sieť v genóme. Rozlišujeme dva typy kódovania podľa toho, nakoľko je zložitý vytvorenie fenotypu na základe informácie uloženej v genotypu. V prípade priameho kódovania (direct encoding) je transformácia genotypu na fenotyp pomerne jednoduchá úloha. Príkladom tohto je kódovanie matice vzťahov, ktorá presne určuje výstavbu neurónovej siete a jednotlivé vzťahy medzi neurónmi s ich intenzitou. Naproti tomu, v prípade nepriameho kódovania (indirect encoding) nie je vzťah medzi genotypom a fenotypom tak zjavný. Príkladom môže byť prípad, kedy sa do genómu priamo neukladajú údaje siete, ale konštrukčné pravidlá, ktorých aplikáciou vzniká fenotyp [23].

Spôsoby kódovania, spolu s ich použitím:

Priame kódovanie (býva tiež označované ako silné alebo nízkoúrovňové):

- na základe spojov (implementované riešenie) – napr. Innervator, Genitor
- na základe uzlov – napr. Shiffmann, Koza
- na základe vrstiev – napr. Genesys, Mandisher
- na základe ciest



Obr. 23 Kódovanie na základe spojov

Nepriame kódovanie (slabé, vysokoúrovňové):

Kódujú sa pravidlá, princíp konštrukcie siete – napr. Kitano, Gruau, Lindenmayer-systems.

Fitness

Na úvod je potrebné zdôrazniť, že počítačové modely, ktoré spájajú neurónové siete s genetickými algoritmami, nepoužívajú pojem fitnessu spôsobom, ktorý je charakteristický pre populačnú biológiu. Kým v druhom prípade je fitness matematicky presne sformulovaný termín, ktorý označuje štatistický parameter vypočítaný z početnosti jednotlivých génov v populácii potomkov, oproti tomu v prvom prípade, pri práve opisovaných modeloch slúži na hodnotenie výkonnosti jedincov, na základe čoho sa rozhodne, či sa genotyp zodpovedný za vznik daného jedinca dostane medzi predkov nasledujúcej generácie. Čiže v tomto ponímaní fitness označuje individuálnu spôsobilosť a skôr sa porovnáva v etológii používaným pojmom „hodnota prežitia“, „hodnota rozmnožovania“, ktorý je menej exaktný, omnoho viac intuitívnych prvkov obsahujúci pojem.

Najjednoduchšia, pritom názorná a často používaná metóda zhodnotenia výkonu modelovaného fenotypu je „binárna funkcia fitness“, ktorá indikuje, či po skončení simulácie jedinec žije, alebo zahynul. Dobré sa aplikuje pri modeloch, kde je cieľom hľadanie potravy. Väčšinou sa rozšíri tým, že sa k modelovanému jedincovi priradí určitá hladina energie, ktorá postupne klesá, no konzumovaním potravy narastie. V prípade, že neurónová sieť, ktorá riadi dané chovanie, nie je dosť úspešná, jedinec ešte pred koncom simulácie zomrie hladom, v opačnom prípade skúšku prežije [14]. V modifikovanej verzii tohto prístupu sa nepočíta energia organizmu, ale množstvo nájdenej potravy, a preto nestačí len prežiť, ale záleží aj na tom, kto našiel najviac

potravu. Samozrejme existujú aj také typy modelov, pri ktorých sa hodnotí aj realizácia nejakých iných úloh, kde kritérium úspešného splnenia nie je také jednoznačné (v implementovanom systéme sú mravce odmeňované napr. za nájdenú potravu, privezenú potravu do mraveniska, za prejdenú vzdialenosť, za nájdený feromón, za jeho intenzitu atď.).

Ekologické neurónové siete

Neurónové siete, ktoré boli vytvorené pre skúmanie regulačných mechanizmov stojacich v pozadí adaptívneho chovania, sa nazývajú niekedy aj ekologickými (econets). Ich podstatou je, že fungovanie siete sa neskúma samostatne, ale priradí sa jej „telo“, ktorého senzory poskytujú vstupy siete, výstupy vznikajúce fungovaním pritom riadia motorický aparát tela (väčšinou ide o 2 motory na bokoch tela, ktoré riadia 2 motoneuróny). Takto vytvorený autonómny agent sa umiestni do prostredia, a tak sa bude činnosť neurónovej siete prejavovať v chovaní jedinca ovplyvneného podnetmi daného okolia. Dôležitou črtou tejto metódy je, že neurónová sieť nielenže spracúva vstupy, ale má možnosť pôsobiť aj na svoje okolie, nakoľko má „telo“. Prostredníctvom svojich senzorov neustále vníma výsledky vlastnej činnosti, ktoré ovplyvnia jej ďalšie chovanie. Týmto vzniká stála spätná väzba, podobne ako pri živých systémoch.



Obr. 24 Prírodný motoneurón

Kapitola 5

Výsledky pokusov

5.1 Experimenty

Pokusy boli vykonané pomocou simulačnej aplikácie (SocInCo), ktorá umožňuje študovať vznik inteligentného chovania a porovnávať rozdielne prístupy riadenia hmyzu. Obsahuje vizualizáciu simulácie, vyhodnocovanie simulácie, porovnávanie, grafické znázornenie úspešnosti kolónie, ich uloženie, zadávanie parametrov simulácie, prostredia. Disponuje dvoma metódami riadenia jedincov: konečný automat a neurónovú sieť v kombinácii s genetickými algoritmami.

Program dovoľuje experimentovať mnohými smermi: chovanie programu ako aj mravcov, prostredia či algoritmov je možné ovplyvňovať prostredníctvom konfiguračného súboru alebo užívateľským dialógom. Tým je možné docieľiť experimenty rôznych druhov, z niekoľkých uhlov pohľadu.

Experimenty (výsledky ktorých pojednáva nasledujúca podkapitola) prebiehali za rovnakých podmienok, aby bolo možné porovnať výsledky. Kolónia má k dispozícii vopred určenú dobu životnosti jednej generácie (nastaviteľná hodnota), počas ktorej pôsobí v prostredí. Simulácia pozostáva z postupnosti takýchto cyklov, výsledky ktorých je možné skúmať ako proces vývoja. Tento prístup je celkom prirodzený v prípade evolvujúcich sa neurónových sietí, kde sa sleduje aj proces učenia, charakteristiky takto vznikajúcej funkcie fitness celého spoločenstva, či vybraného jedinca. No tieto za sebou nasledujúce generácie dávajú presnejšie výsledky i pri automatoch, ich výsledky sa dajú štatisticky spracovať (i keď sa ich fitness v dôsledku absencie evolúcie nijak dramaticky nemení, mal by kolísť okolo určitej hladiny).

Spomínané podmienky prostredia boli vo väčšine prípadov nasledovné:

- počet mravcov: 18
- počet potravy (jednotlivé typy): 50, 100, 150
- minimálna vzdialenosť od mraveniska: 100
- pozícia mraveniska: stred prostredia

Ostatné parametre sa menili podľa prístupu, alebo podľa cieľa toho ktorého experimentu.

Pokusy boli zamerané hlavne na porovnanie rôznych prístupov riadenia jedincov a na skúmanie toho, aký mala prínos tá ktorá metóda a ktorá je z nich viacej realistická resp. efektívnejšia. Boli vykonané i experimenty týkajúce sa vplyvu rôznych zmien vstupných parametrov simulácie (parametrov prostredia, jedincov či ich riadiacich algoritmov).

Skúmali sa aj rôzne architektúry neurónových sietí, ale ukázalo sa, že na presnom počte neurónov či dokonca vrstiev až tak nezáleží. Tieto by začali hrať väčšiu rolu v prípade už suboptimálne nastavených parametrov, po fine-tuningu samotných sietí a genetických algoritmov, resp. celého systému ako takého.

5.2 Vyhodnotenie pokusov

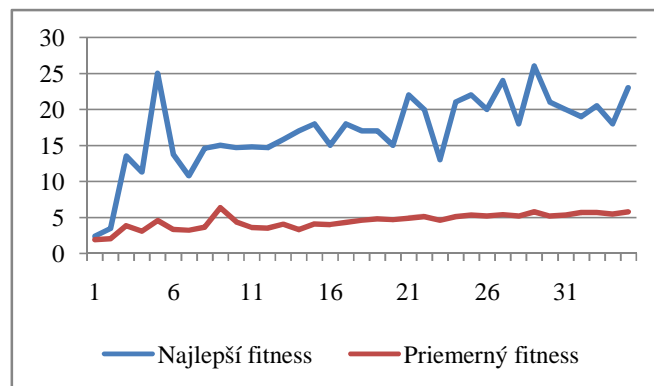
Po mnohých testoch a experimentoch je možné tvrdiť, že mravce riadené konečným automatom boli úspešnejšie, ako tie riadené neurónovou sieťou (či už počtom dovezenej potravy alebo čisto vizuálnym dojmom). Mohlo to byť zapríčinené tým, že automat sa lepšie hodí na modelovanie tak jednoduchého jedinca, nakoľko i on sám je prostou výpočtovou jednotkou. No určite to bolo i tým, že neurónové siete a genetické algoritmy sú veľmi citlivé na presné nastavenie ich parametrov a metódy tohto typu ľahko môžu uviaznuť v rôznych lokálnych minimách. No problémy môžu byť aj s ich pomalou konvergenciou, nie náhodou boli v minulosti významnejšie aplikácie tohto charakteru vyvíjané na výkonných paralelných prostriedkoch. Nie je však priestor nechať bežať evolúciu na stolných počítačoch tisíce generácií.

K podobnému záveru dospeli aj Jefferson a Collins (tvorcovia Genesys), ktorý porovnali mravce založené na NN a FSM a vyvodili dôsledok, že mravce NN sú výpočtovo o niečo slabšie ako FSM mravce.

Neurónové siete

Experimenty týkajúce sa neurónových sietí prebiehali skôr s účelom skúmania učenia, evolúcie, ako samotného vytvoreného chovania. Bolo to kvôli tomu, že sa prístup automatov osvedčil viac, predstavoval vernejší model, ktorý sa lepšie skúmal z hľadiska analógií s prírodou.

Výsledky simulácií, procesu učenia ukázali, že sa úspešnosť skúmaného typu chovania počas generácií zvyšovala, no miera stúpania nebola rovnomerná. Rast bol pri prvých generáciách najprudší, následne stúpala úspešnosť už len veľmi pomalým tempom, až kým nezačala stagnovať a kolísať okolo jednej hodnoty. Tento vývoj fitnessu zachycuje graf na obrázku č. 25. Bol to konkrétny experiment trvajúci 35 generácií, každá trvala 200 sekúnd, s nízkenou minimálnou vzdialenosťou potravy od mraveniska (boli vykonané i pokusy dlhšieho trvania, no vývoj fitnessu sa radikálne nezmenil).



Obr. 25 Vývoj fitnessu neurónových sietí (35 generácií, každá trvala 200 s)

Proces učenia je tam badateľný, vychádzajúc nie len z priebehu funkcie fitness, ale i z prejavov chovania mravcov (čo robia, ako to robia). V súvislosti s vykonanými simuláciami existuje viacero momentov, ktoré stoja za zmienku.

Podľa očakávania bola pri každom pokuse počiatočná generácia života neschopná. Mravce krúžili na mieste, vôbec neopúšťali okolie mraveniska. Výnimkou bolo len zopár jedincov, ktorí sa vydali celkom rovno, bez toho, aby akokoľvek menili svoj smer – nakoniec narazili na hranice prostredia a nevedeli ako ďalej.

Bolo evidentné, že takto nedosiahnu nijaké výsledky, veď vôbec neboli schopní pohybovať sa rozumne po prostredí. Po niekoľkých generáciách (väčšinou okolo 150) nastali nasledujúce zaujímavosti (aj keď jednotlivé prvky chovania nie sú tak výrazne viditeľné, ako u automatov, no sú určite nádejné):

- vzdialili sa od mraveniska, putovali cez celé prostredie
- našli potravu, čo je dosť podstatné
- vedeli sa pohybovať oveľa rýchlejšie, lepšie si poradili aj s hranicami prostredia
- používali feromón, a to skoro vždy – je to zaujímavé, lebo to nemali nijak explicitne určené, aby tak robili. Tú schopnosť sa museli naučiť a používať feromón v prípadoch, keď to „považovali za vhodné“.
- poniektorí feromón aj nasledovali – to bolo asi najväčšie prekvapenie zo všetkých
- mnohí z nich aj zosilňovali nájdený feromón, čo im tiež nikto nekázal
- výsledkom zlepšeného pohybu sa občas aj vrátili do mraveniska

Treba podotknúť, že väčšinou mali na učenie veľmi málo času, aby bolo možné zrealizovať viacero generácií. Krátko na to, ako sa narodili, už prišla nová generácia (životnosť väčšinou v rádoch minút).

Kľúčovú rolu hrá i systém odmien mravcov, nakoľko od fitnessu závisí, akým smerom sa bude sieť vyvíjať, aké situácie budú mravce preferovať (v aplikácii sa dáva odmena napr. pre dovezenie potravy, nájdenie potravy a feromónu v závislosti na jeho intenzite, za prejdenú vzdialenosť apod. – výška hodnôt je nastaviteľná).

Na stránkach projektu (kópia je súčasťou CD prílohy) sa nachádzajú aj video zábery počiatočných pokusov s neurónovými sieťami.

Jedným zo spôsobov prekonania nedostatkov siete by bolo aj zvýšenie počtu vstupov, tak by mala neurónová sieť väčšie množstvo informácií podporujúcich rozhodnutia.

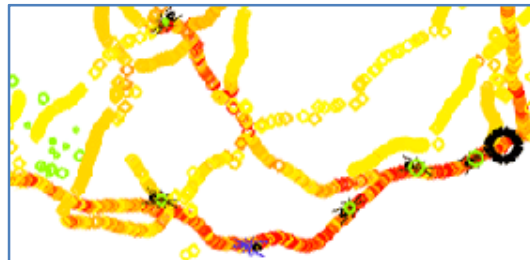
Konečný automat

V prípade konečného automatu bolo najväčšou výhodou to, že sa mravce nemuseli učiť všetko „od začiatku“, od chôdze, až po používanie, nasledovanie feromónu. Pri tomto spôsobe riadenia sú úspechom iné momenty, ako napríklad to, že sa mravce nielenže vracajú s potravou do mraveniska (čo je veľký úspech,

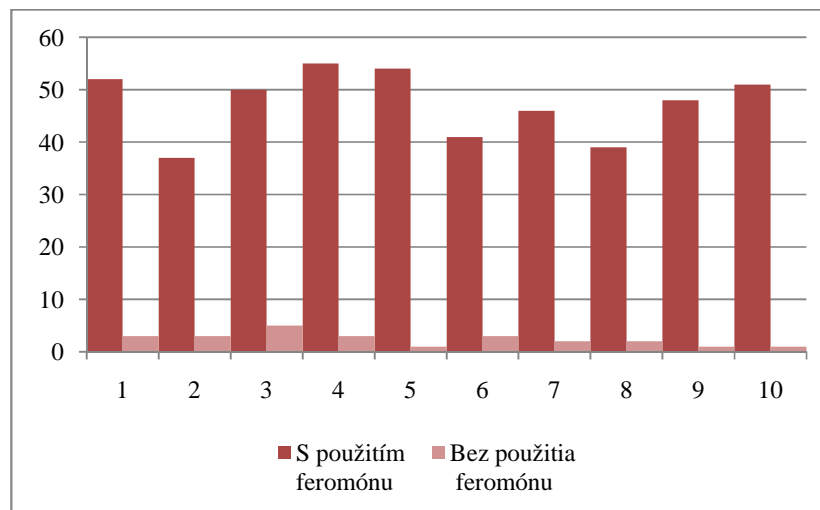
nakol'ko to nie je celkom jednoduchá úloha), ale aj zvyšujú vo veľkej miere produktivitu svojho snaženia, vytvárajú cestičky, ktoré potom využívajú a zosilňujú.

Ukázalo sa, že hľadanie potravy bez feromónu je v priemere (cez rôzne časy životnosti generácií, ktoré boli dosť dlhé na to, aby boli štatisticky významné) až o 95% menej efektívne (čo sa týka počtu dovezenej potravy, ergo úspešnosti kolónie). Absencia kooperácie znamená chaotické chovanie, s náhodnými objavmi mraveniska. Potravu síce našli, ale mravenisko, cestu späť už nájst' nevedeli (pri náhodnom putovaní je menšia šanca nájst' mravenisko ako na väčšom území rozprestretú potravu). Obrázok č. 27 znázorňuje vývoj počtu dovezenej potravy pre jednu pevne danú hodnotu času (600 s) pôsobenia generácie (opakované spustenie).

Tento experiment potvrdil vznik určitého stupňa inteligencie, nakol'ko pomocou vytvoreného chovania sa s veľkou mierou zvýšila i efektivita kolónie. Táto forma vzoru vyššej organizácie bola pozorovateľná i na samotnom počínaní jedincov, nie len na číselných výstupoch aplikácie (príklad javu je na obr. 26). Vytvorili sa feromónové stopy rôznych stupňov intenzity, mravce postupne začali využívať len zopár najsilnejších, ktoré takto ešte viac získali na sile. Vytvorilo sa kolektívne chovanie.



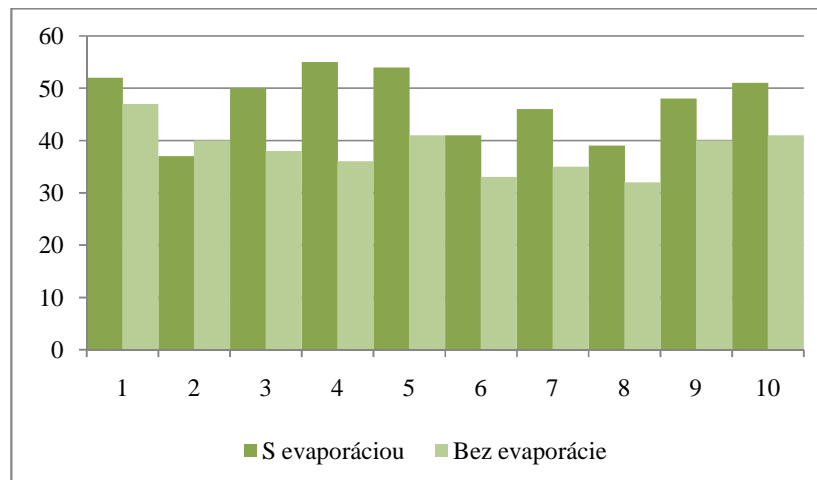
Obr. 26 Vznik silnejších feromónových ciest (intenzita červenej farby odpovedá sile feromónu, modrý jedinec „je na stope“ smerom k potrave, čierny objekt je vchod mraveniska, zelené objekty sú potrava)



Obr. 27 Vývoj počtu dovezenej potravy v závislosti na používanie feromónu (10 generácií)

Ďalší pokus sa zamerával na evaporáciu feromónu a na jej vplyv na úspešnosť systému. Využil z časti výsledky pokusov z predchádzajúceho bodu. Tie, ktoré sa týkali hľadania potravy za pomoci feromónu, ktorý postupne slabol, ak nebol používaný. S týmito výsledkami sa porovnali výsledky získané z behov simulácií, v ktorých sa vyplo vyparovanie feromónu., čiže ten celú dobu len pribúdala. Počet

potravy dovozenej do mraveniska klesol (obr. 28). Môžeme predpokladať, že sa to udialo následkom zmätenia mravcov, nakoľko feromónu bolo vždy viac a viac, a preto už nepredstavoval takú dobrú a nápomocnú informáciu. Cesty nepotrebné, menej využívané nezmizli, predstavovali tak faktor zmätenia.



Obr. 28 Vývoj počtu dovezenej potravy v závislosti na evaporácii feromónu (10 generácií)

Kapitola 6

Záver

6.1 Konklúzia a myšlienky budúcej práce

Napriek tomu, že sú mravce stvorenia tak jednoduché disponujúce nízkou kognitívnou kapacitou, kolektívne telo ich kolónií je schopné riešiť komplexné úlohy. Ich masívna interakcia spôsobuje, že je ich chovanie emergentné. Práve toto chovanie už inšpirovalo mnohých výskumníkov z oblasti informatiky.

Je potrebné podotknúť, že umelá inteligencia, ako aj samotná počítačová veda sú stále len v počiatočných vývoji v porovnaní s pokrokmi iných vedných odborov. Ešte novšou oblasťou je štúdia umelých mravcov, kde odborníkov čaká ešte veľký kus práce.

Na poli umelej inteligencie existuje veľké očakávanie, že emergentné systémy pomôžu vyplniť medzeru, ktorá ostala za tradičným, na symboloch založeným prístupom, kde je učenie a vysporiadanie sa s novými situáciami vždy veľká výzva. Výskumníci dúfajú, že im mravčie algoritmy v budúcnosti pomôžu prekonať tento problém.

Po štúdiu tematiky a pokusoch je zjavné, aký je feromón užitočný, ako môže zefektívniť hľadanie potravy a ako môže byť použitý tento koncept pri výskumoch informatického charakteru, špeciálne na poli sietí (neurónových) a umelej inteligencie.

Tak, ako i ostatné štúdie o mravčích algoritmoch, i táto práca používa k riadeniu mravcov výhradne koncept feromónu, čo občas môže viesť ku katastrofickému chovaniu, nakoľko mravce sa pri navigácii v prírode nespoliehajú výhradne na feromón. Mohlo by byť zaujímavé vyvinúť systém kolónie jedincov, ktorý by zahŕňal i ďalšie prostriedky biologických mravcov, ako napr. gravitáciu, svetelné zdroje či zrak.

Nápady do budúcnosti

Čo sa týka riadenia jedincov, evolvujúce konečné automaty sa ukazujú ako zaujímavý koncept. Konečné automaty by mohli byť reprezentované ako indivídium v genetickom algoritme vo forme série prechodov. Mohol by sa kódovať spôsob tvorby automatu, princíp konštrukcie, čím by stavy mohli pribúdať, ubúdať, mohli by sa krížiť celé časti automatov, v prospech vytvorenia toho „ideálneho“.

Ďalším možným nápadom v prípade FSM je pridanie memórie. Bolo by zaujímavé sledovať vplyv jej veľkosti na chovanie jedinca, ako aj na chovanie celého spoločenstva, ako celku.

Lákavou myšlienkou by bolo pustiť do prostredia jedince s rôznymi typmi riadenia, ako napr. evolvujúce sa neurónové siete či automaty. Je otázka, čo by to mohlo priniesť, aké výhody, akým spôsobom by prebiehala spolupráca či interakcia.

Pri implementácii mravcov by bolo určite zaujímavé skúsiť vytvoriť rôzne role. Povedzme mravce, ktoré hľadajú potravu, mravce transportné (spotrebovali by viacej energie), apod. Mohol by sa skúmať dopad zmien pomerov jednotlivých typov jedincov na výsledky. Tieto zmeny by mohli hrať rolu i pri pokusoch cieľavedome zvýšiť úspešnosť celej kolónie ako celku.

V prípade prostredia sa naskytá taktiež niekoľko sľubných nápadov. Jedným z nich je editor prostredia, v ktorom by bolo možné postaviť taký svet, kde by sa odohrávala celá simulácia. Editor by sa dal rozšíriť aj o interaktívne zmeny počas behu modelovania. Tým by bolo možné skúmať rôzne vplyvy zmien, schopnosť jedincov ako aj celého kolektívneho spoločenstva prispôbovať sa náhlým modifikáciám okolia (napr. umiestenie bremena cez už vytvorenú feromónovú stopu).

V tomto projekte som sa pokúšal vytvoriť nástroj pre štúdium rôznych prístupov riadenia sociálneho hmyzu, ako aj pochopiť ich samotné správanie a pozadie ich úspechu. Vychádzajúc z experimentálneho a výskumného charakteru práce, má kód aplikácie samozrejme ešte dosť priestoru na zlepšenie a pokrok. Najväčším problémom pri aplikácii bola realizácia mozgových jednotiek mravcov za podmienky, že majú len jediný spôsob riadenia jedincov – prostredníctvom dvoch „pásov“ na ich bokoch. Nemenej náročné bolo i presné nastavenie obrovského počtu parametrov (to hlavne u neurónových sietí a genetických algoritmov), tento problém sa kvôli jeho podstate ani nedá dokonale vyriešiť.

Hlavný prínos práce spočíva v dôkaze možnosti aplikácie biologicky motivovaných výpočtových metód pri skúmaní kolektívneho emergentného chovania. Taktiež vo výskume v prírode úspešne fungujúcich analogických riešení ako aj informatikmi už vytvorených podobných systémov. Ústredným výsledkom práce je spozorovaný fakt, podľa ktorého sa konečný automat pre študovaný problém hodí viac, ako evolvujúca sa neurónová sieť. No nemenej prínosné boli i výsledky pokusov zaoberajúce sa úlohou feromónu v prípade vzniku inteligentného kolektívneho chovania. Pokusy s neurónovými sieťami zas naznačili, že možné riešenie by mohlo viesť i týmto smerom, ale automaty sa nateraz pri tomto type simulácií zdajú byť vhodnejšou alternatívou. Ohromný priestor prehľadávania pojednávaných neurónových sietí je veľká výzva i v prípade optimálneho nastavenia veľkého počtu parametrov (ktorých nastavenie je podobne náročný problém). Veľkosť univerza pri pokusoch a testoch bola omnoho väčšia, ako v systémoch podobného charakteru, nakoľko svet mravcov bol realizovaný ako kvázi spojitý (za účelom priblíženia sa realite), oproti v tomto obore dominantnému diskrétnemu prostrediu, hrubej mriežky a extrémne zjednodušeným mravcom. V prípade týchto zjednodušených modelov je otáznne, nakoľko je možné považovať vyvedené dôsledky za adekvátne pôvodnému zámeru.

V niektorých kapitolách som sa opieral o poznatky ľudí, ktorí venovali celý svoj profesionálny život tejto zaujímavej oblasti vedy – myrmekológii (veda zaoberajúca sa mravcami), nakoľko sa väčšinou zanedbáva práve biologický aspekt problematiky, ktorý je základom celej myšlienky.

Je dokázané, že keď sa biologicky inšpirované myšlienky aplikujú na poli umelej inteligencie, majú veľkú šancu na úspech. Môžeme sa mnoho naučiť od prírody, ktorá pomocou osvedčených metód, bez námahy rieši každodenné problémy a možno raz i my v plnej miere využijeme tieto metódy v prípade strojov, ktoré sme vytvorili a necháme sa poučiť od toho, kto to mal už dávno pred nami dokonale vymyslené – od prírody.

Detailné vypracovanie a implementácia pojednávaných prístupov sa môže zdať ambicióznym cieľom, no dnes už mnohí pracujú na realizácii takých simulačných programov, ktoré sa zaoberajú niektorým zo spomínaných aspektov. Výsledkom rapídneho vývoja počítačových vied sa otvorili cesty k vytvoreniu a výskumu takých modelových systémov, ktoré boli pred jedným či dvoma desaťročiami zaradované do oblasti vedecko-fantastickej.

Literatúra

- [1] Bertalanffy L. Von (1962): Modern Theories of Development: An introduction to theoretical biology. Harper, New York.
- [2] Mitchell M. (1993): Computer Models of Complex Adaptive Systems. In New Scientist.
- [3] Csányi V. (1988): Evolutionary Systems. General theory of evolution. Gondolat, Budapest.
- [4] Mařík V., Štěpánková O., Lažanský J. a kol. (2003): Umělá Inteligence (4). Academia, Praha.
- [5] Hillis W. D. (1985): The Connection Machine. MIT Press, Cambridge.
- [6] Franklin S. (1997): Autonomous Agents as Embodied AI (<http://www.msci.memphis.edu/~franklin/AAEI.html>)
- [7] Ray T. S. (1994): An evolutionary approach to synthetic biology: Zen and the art of creating life. In Artificial Life 1(1/2): 195-226. MIT Press.
- [8] Hoelldobler B., Wilson E. O. (1997): Cesta k mravencům. Academia, Praha.
- [9] Brewer G. (2000): Social insects. (<http://www.ndsu.nodak.edu/entomology/topics/societies.htm>)
- [10] Turing A. M. (1950): Computing Machinery and Intelligence. In Mind 49: 433-460. Oxford University Press.
- [11] Searle J. R. (1990): Is the Brain a Digital Computer? In Scientific American, Vol. 262, No. 1: 26-31.
- [12] Searle J. R. (1980): Minds, Brains, and Programs. In The Behavioral and Brain Sciences 3: 417-24.
- [13] Boden M. A. (1988): Escaping from the Chinese Room. In Computer Models of Mind, Chapter 8. Cambridge University Press.
- [14] Beer R. D. (1990): Intelligence as Adaptive Behavior. An Experiment in Computational Neuroethology. Academic Press, San Diego.
- [15] Taylor C., Jefferson D. (1994): Artificial Life as a Tool for Biological Inquiry. In Artificial Life 1/2: 1-13. MIT Press.
- [16] Miller G. F. (1995): Artificial Life as Theoretical Biology : How to do real science with computer simulation. Cognitive Science Research Paper 378, School of Cognitive and Computing Sciences, University of Sussex.
- [17] Steels L. (1995): The origins of intelligence. In Proceedings of the Carlo Erba foundation meeting on artificial life, Milano.
- [18] Mařík V., Štěpánková O., Lažanský J. a kol. (2003): Umělá Inteligence (3). Academia, Praha.
- [19] Langton Ch. G. (1986): Studying Artificial Life with Cellular Automata. Physica D 22: 120-149.

- [20] Jefferson D., Collins R., Cooper C., Dyer M., Flowers M., Korf R., Taylor C., Wang A. (1991): Evolution as a Theme in Artificial Life: The Genesys/Tracker System. In Artificial Life II: 549-578, Addison-Wesley.
- [21] Drogoul A., Corbara B, Fresneau D. (1995): MANTA : New experimental results on the emergence of (artificial) ant societies. University of London College Press, London.
- [22] Hart W. E., Belew R. K. (1991): Optimizing an arbitrary function is hard for Genetic Algorithms. In Proc. Fourth Intl. Conf. on Genetic Algorithms: 190-195. Morgan-Kaufman, Los Altos.
- [23] Balakrishnan K., Honavar V. (1995): Evolutionary Design of Neural Architectures – A Preliminary Taxonomy and Guide to Literature. Technical report CS TR#95-01, Artificial Intelligence Research Group.
- [24] Whitley L. D., Gruau F. (1993): Adding Learning to the Cellular Development of Neural Networks – Evolution and the Baldwin effect. In Evolutionary Computation 1(3): 213-233.

- [D] Ants – Nature’s Secret Power (2004), Bert Hoelldobler's documentary film. Written, directed & photographed by Wolfgang Thaler.

Dodatok

Obsah priloženého CD

Na priloženom CD nájdete:

- Zdrojové kódy programu (zložka **Source**)
- Spustiteľnú verziu programu (zložka **Binary**)
- Kópiu stránok projektu - <http://www.ulman.us/projects/socinco/> (zložka **Web**)
- Text tejto práce (zložka **Text**)

Spustenie aplikácie:

- Skopírujte zložku **Source** na svoj pevný disk, nie je potrebná inštalácia
- Spustíte súbor **SocInCo.exe**

Prílohy

Programátorská a užívateľská dokumentácia

SocInCo – Vývojová dokumentácia

Autor: Attila Ulman (ulman.attila@gmail.com)

Vedúci práce: RNDr. Pavel Surynek



Názov:

Vznik inteligentného chovania u hmyzích spoločností

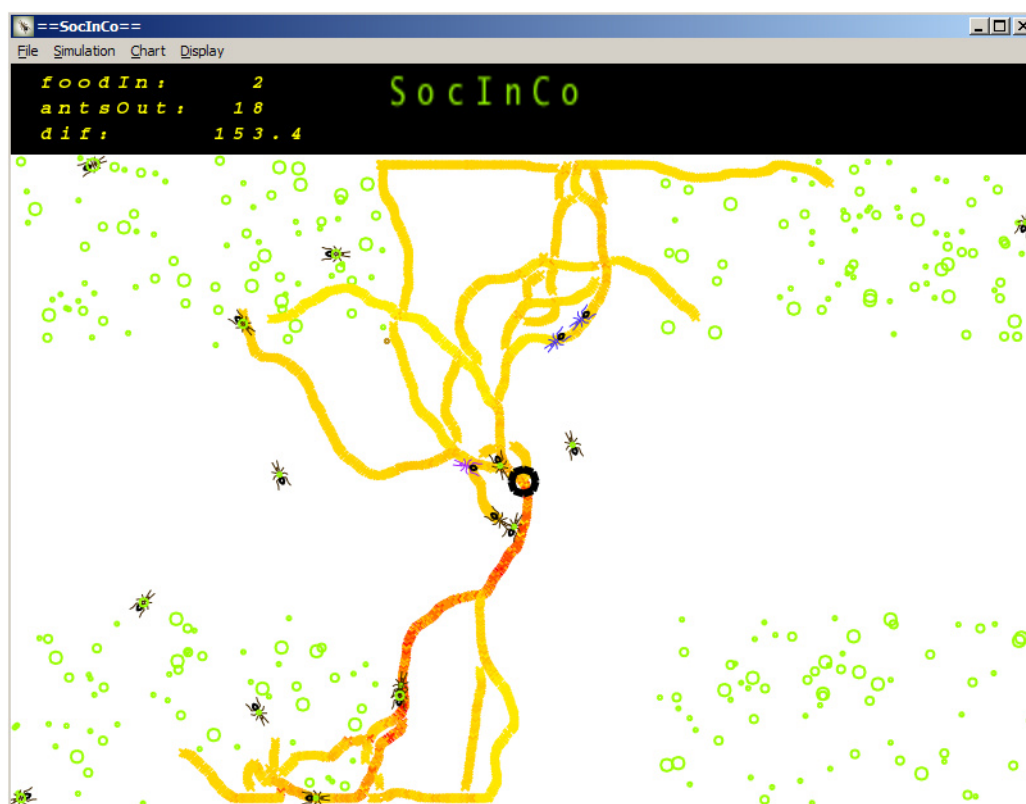
Platforma:

Windows

Jazyk a prostredie:

C++, Visual Studio

Samotná aplikácia – prostredie pre simuláciu (SocInCo). Obsahuje vizualizáciu simulácie, vyhodnocovanie simulácie, porovnávanie, grafické znázornenie vybraných parametrov/úspešnosti kolónie, ich uloženie, zadávanie parametrov simulácie/prostredia. Implementované boli dve metódy riadenia jedincov: automat a neurónová sieť.



Screenshot prostredia

1 Popis logiky programu

Program beží na dvoch úrovniach. Nad samotnou výpočtovou vrstvou sa nachádza vizualizačná vrstva, ktorá sa stará o zobrazovanie prebiehajúcej simulácie. Pre obe tieto vrstvy vytvára prostredie WinApi, ktoré zabezpečuje správu okien, rôzne dialógy, menu a podobne.

1.1 Výpočtová vrstva

Jadro výpočtovej vrstvy tvorí „riadiaca centrála“ celej simulácie. Tá obhospodaruje celé prostredie, riadi jednotlivé cykly simulácie, dohliada na celý systém „zhora“. Na túto sú napojené ostatné moduly, najdôležitejší je modul jedincov v prostredí (je tam ešte napr. modul objektov prostredia, modul parametrov, logovací modul, modul timeru atď.).

Modul jedincov:

(Ant.cpp, Ant.h; AutoBr.cpp, AutoBr.h; NeuralBr.cpp, NeuralBr.h; Neuron.cpp, Neuron.h; NeuronLay.cpp, NeuronLay.h; GeneticA.cpp, GeneticA.h; Genome.cpp, Genome.h)

Jedná sa tak isto o dosť komplexný modul obsahujúci ďalšie a ďalšie.

Základná trieda pre mravce je BaseAnt.

```
// ... // časť z Ant.h
class BaseAnt;

class Brain {
public:
    virtual void Update ( BaseAnt* ThisAnt ) abstract;
};

class BaseAnt {
public:
    BaseAnt ( Brain* );
    void Update();
    bool ChckNearestCol ( double NPosX, double NPosY,
                        double NearSize );
    bool ChckPher ( vector<Pheromone> Pher );
    void SetFoodType ( int type );
    void UnsetFood ();
    Pheromone LayPher ();
    void HitWall ();
    void IncFit ( double val );

    double Get_PosX () const { return m_PosX; }
    double Get_PosY () const { return m_PosY; }
    double Get_Rotation () const { return m_Rotation; }
    bool Get_Carry () const { return m_Carry; }
    int Get_CarriedType () const { return m_CarriedType; }
    double Get_Size () const { return m_Size; }
    int Get_ClosestPher () const { return m_ClosestPher; }
    double Get_Fitness () const { return m_Fitness; }

    GLubyte m_Color[3];

    friend class AutomatonBrain;
    friend class NeuralBrain;
```

```

private:
    Brain* m_Brain;
    double m_PosX, m_PosY, m_PrevPosX, m_PrevPosY;
    double m_DirX, m_DirY;
    double m_Rotation;
    double m_Speed;
    int m_ClosestPher;

    double m_RCaterpil, m_LCaterpil;

    bool m_Carry;
    int m_CarriedType;
    double m_Size;
    double m_Fitness;
};
// ...

```

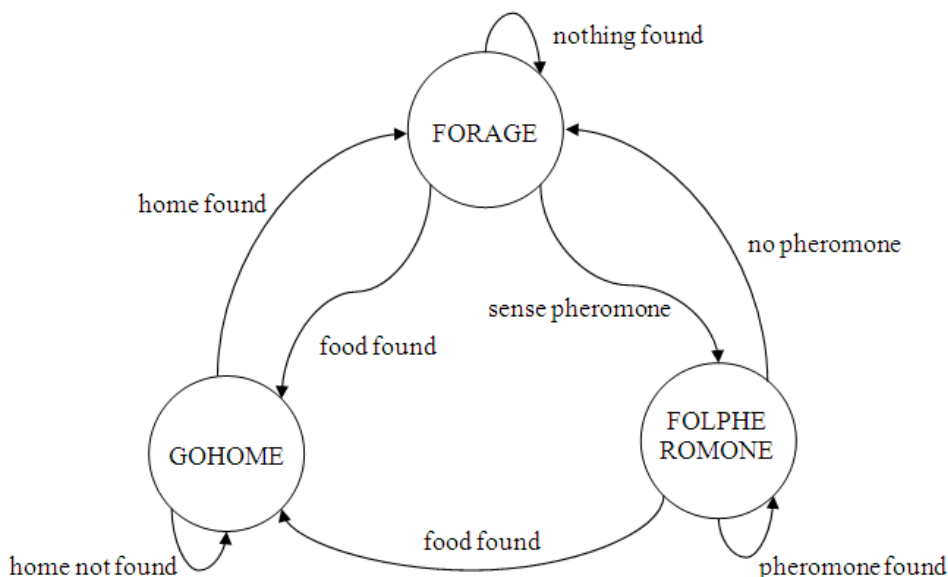
Každý mravec má zásuvku, do ktorej je možné zasúvať rôzne typy riadiacich jednotiek. On ich používa cez jednotný interface, ktorý poskytuje základná trieda mozgu – Brain – a virtuálna dedičnosť. Každý jedinec má pointer na svoj riadiaci modul a tabuľka virtuálnych funkcií zabezpečí volanie tých správnych metód.

Boli implementované dve verzie mozgových modulov: automat a neurónová sieť. Jediný nástroj pre prenášanie informácií z mozgu je riadenie síl na pásoch jedinca (popísané nižšie).

- **Automat** (AutoBr.cpp, AutoBr.h):

Komplexný problém sa rozčlení na niekoľko podproblémov. Keďže sa agenti riadia jednoduchými pravidlami, je možné implementovať ich chovanie konečným automatom.

Konečný automat má nasledujúce stavy a prechody medzi nimi:



Mravce sa narodia do stavu: hľadanie potravy a začnú svoju „náhodnú“ púť za potravou. Keď sa dostanú do kolízie s hranicami prostredia, otočia sa a pokračujú v hľadaní. Keď narazia na potravu, vezmú ju a prejdú do stavu späť s potravou domov. Po zložení potravy v mravenisku sa vrátia do stavu:

hľadanie potravy. Keď mravec zdetekuje feromón, prepne sa do stavu: nasledovanie feromónu.

(možné je i tento spôsob prepojiť s genetickými algoritmami aby sa automat čo najviac vyvinul a pracoval čo najefektívnejšie)

Pri detekcii feromónu je potrebné ho nasledovať do určitej miery. To zabezpečuje tzv. „steering model“. Hlavnou úlohou tohto modelu sú výpočty uhlov pri nasledovaní feromónu mravcami. Je dôležité, aby si dobre poradili s feromónovými cestami, a tak mohli prinášať oveľa viac potravy do mraveniska.

Počítanie uhlov sa robí tak, že sa spočíta vektor feromónu od mravca, ktorý sa normalizuje. Skalárny súčin s vektorom mravca dá uhol medzi nimi. Podľa toho sa rozdeľujú potom sily na jednotlivé strany - v závislosti na ich vzájomnej polohe.

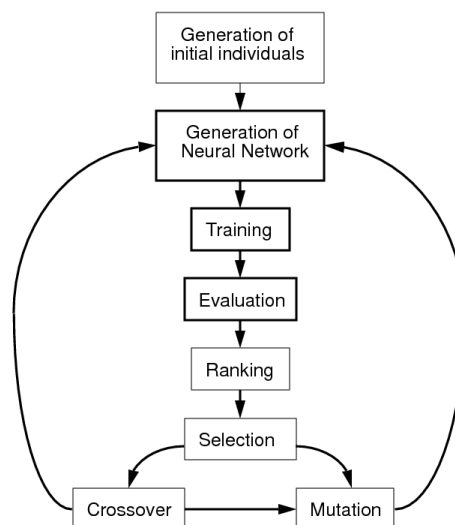
V tých správnych stavoch sa kontroluje kolízia s mraveniskom, feromónom, potravou, zvyšuje/znižuje sa fitness, počítajú sa sily, príp. sa vylučuje feromón a podľa daných špecifik sa potom prechádza medzi stavmi.

```
// ... // časť z AutoBr.cpp
void AutomatonBrain::Update ( BaseAnt* ThisAnt )
{
    switch ( m_State ) {
        case FORAGE:
            Forage( ThisAnt );
            break;
        case GOHOME:
            GoHome( ThisAnt );
            break;
        case FOLPHEROMONE:
            FolPheromone( ThisAnt );
            break;
    }
}
// ...
```

- **Neurónová sieť** (NeuralBr.cpp, NeuralBr.h; Neuron.cpp, Neuron.h; NeuronLay.cpp, NeuronLay.h; GeneticA.cpp, GeneticA.h; Genome.cpp, Genome.h):

Neurónové siete v spojení s genetickými algoritmami – siete sa učia genetickými algoritmami – evolvujú sa samotné siete.

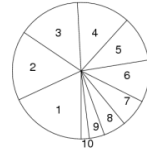
Celý vývoj sietí je ponechaný na genetických algoritmoch. Je to kvôli tomu, že nie je možné učenie klasickými metódami. Chýbajú nám správne výstupy siete. Do evolúcie vstupujú chromozómy mravcov, ktoré sú reprezentované váhami ich sietí. Tieto sa podrobujú selektívnej reprodukcii (na základe fitnessu –



nájdená resp. privezená potrava, vzdial. atď.). Následne prichádza selekcia, kríženie a mutácia.

Selekcia:

Použitá bola ruletová selekcia – silnejší jedinci majú väčšiu šancu na výber. Bol uplatnený tiež elitizmus – niekoľko najlepších jedincov sa bez zmeny okopíruje do novej generácie, aby sa doteraz najlepší nestratili pri genetických zmenách.



Kríženie:

Používa sa dvojbodové kríženie (s náhodným výberom bodov)

Parent 1:	001010011	01010010	10101110
Parent 2:	010101110	10101011	01110101
	▼	▼	▼
Child:	001010011	10101011	01110101

Mutácia:

Mutuje sa s danou pravdepodobnosťou (okolo 0.2) - šanca pre vznik nových typov jedincov.

Fitness – kritérium výkonnosti - mravce sú odmeňované napr. za nájdenú potravu, privezenú potravu do mraveniska, za prejdenú vzdialenosť, za nájdený feromón, za jeho intenzitu atď. Od fitnessu závisí, akým smerom sa bude sieť vyvíjať, aké situácie budú mravce preferovať (vysoká variabilita odmien). Takto evolvované siete riadia pohyb mravcov a vylučovanie feromónu. Ako vstupy dostávajú napr. pozíciu mravca, jeho rotáciu, či nesie potravu, akú vzdialenosť prešiel od posledného update-u, apod.

Kódovanie sietí je priame – na základe spojov.

Niektoré konkrétne parametre implementovanej siete:

Neurónová sieť:

„feed forward“ sieť – spojenie typu „full connection“.

počet skrytých vrstiev: 1 (6 neurónov)

počet vstupov: 5

počet výstupov: 3 (2 pásy po bokoch mravca, feromón)

prenosová funkcia: sigmoida

Genetický algoritmus:

elitizmus: 4 najlepší jedinci rovno do ďalšej generácie.

pravdepodobnosť mutácie: 0.15

miera mutácie: 0.3

životnosť 1 generácie: 1000

odmeny (napr.) 7.0 (dovezená potrava)

2.0 (nájdená potrava) a ďalšie.

(celá štruktúra od počtu vrstiev, neurónov až po konštanty siete je nastaviteľná v konfiguračnom súbore)

Ďalšia štruktúra mravcov - majú len malú plochu v priestore očí, ktorá je analogická jednému pixlu na obrazovke. Je to malý rozsah vnímania či detekcie, čo simuluje ich slabý zrak. Tento rozsah využívajú jak na detekovanie objektov,

tak aj na vnímanie feromónu. Každý mravec je vybavený 3-3 nohami na oboch stranách tela, ktoré sa môžu brať ako dva pásy (analogické tým, ktoré majú tanky) na oboch stranách. Tieto behúne emulujú šesticu nôh mravca, umožňujúc tak lezenie po prostredí. Mozog mravca má tieto behúne jako jediný spôsob ich riadenia. Výsledok každého typu mozgu sú sily na oboch stranách, z ktorých sa následne počíta rotácia, rýchlosť a pozícia jedinca. Toto spracovanie výstupov mozgu sa deje v každom cykle:

```
// ... // časť z Ant.cpp
void BaseAnt::Update ()
{
    m_Brain->Update( this );

    double rFrc = m_LCaterpil - m_RCaterpil;
    rFrc = Lim( rFrc, g_Param->m_minTurnTolerance,
               g_Param->m_maxTurnTolerance );
    m_Rotation += rFrc;
    m_Speed = 1.5 * ( m_LCaterpil + m_RCaterpil );
    if ( m_Carry )
        m_Speed /= 1.5;
    m_DirX = -sin( m_Rotation );
    m_DirY = cos( m_Rotation );

    m_PrevPosX = m_PosX;
    m_PrevPosY = m_PosY;
    m_PosX += m_DirX * m_Speed;
    m_PosY += m_DirY * m_Speed;
    HitWall();
}
// ...
```

Mravce majú schopnosť vylučovať feromón: slúžiaci na indikáciu cesty – stopa, ktorú zanechávajú keď blúdia prostredím a hľadajú potravu, pomocou ktorej sa vytvára kooperácia medzi jedincami.

Feromón uvidia len vtedy, ak je pred nimi v dostatočnej vzdialenosti a je v určitom výseku:

```
// ... // časť z Ant.cpp - z ChckPher
if ( len < g_Param->m_PherSize + m_Size )
{
    double normPherX = dstX / len;
    double normPherY = dstY / len;
    int rotAngle = -( (int)RADTODEG( m_Rotation ) - 270 ) % 360;
    if ( rotAngle < 0 )
        rotAngle = 360 + rotAngle;
    double normAntX = cos( DEGTORAD( rotAngle ) );
    double normAntY = sin( DEGTORAD( rotAngle ) );
    if ( RADTODEG( acos( normPherX*normAntX + normPherY*normAntY ) )
        < 90 )
    {
        if ( m_Carry )
        {
            m_ClosestPher = i;
            return true;
        }
        if ( len < minLen )
        {
            minLen = len;
            actMin = i;
        }
    }
}
// ...
```

Ďalej je možné spýtať sa mravca, či koliduje s najbližším daným objektom (ChckNearestCol), dať mu určitý druh potravy (SetFoodType), zobrať mu ju (UnsetFood), povedať mu, aby vylučoval feromón (LayPher), aby si zvýšil hodnotu fitness (IncFit), je tam taktiež ošetrovanie hraníc prostredia (HitWall).

Mravce sú situované do 2D ohraničeného prostredia, z ktorého nemôžu vypadnúť ani ho opustiť. Mravce nekolidujú s ostatnými, ale preliezajú cez seba. Prostredie obsahuje potravu viacerých veľkostí, ktoré je rozmiestňovaná náhodne. Potrava sa samozrejme generuje v určitej vzdialenosti od mraveniska, aby sa prejavilo putovanie za potravou (nastaviteľná vzdialenosť). Táto metóda simuluje realistickejšie hľadanie ciest k zdroju potravy. Keď sa mravec dostane do kolízie s potravou, vezme potravu so sebou a pokúsi sa ju odnieť do mraveniska.

Radiaca centrála:

(HQuarters.cpp, HQuarters.h; HQuartersAuto.cpp, HQuartersAuto.h; HQuartersNN.cpp, HQuartersNN.h)

Funguje na princípe virtuálnej dedičnosti a abstraktných funkcií. Celá radiaca logika je celkom obecná a výmenná, s jednotným rozhraním pre volanie.

Služby rovnaké pre všetky typy centier sú volané cez spoločného predka, tie špecifické pre daný typ mozgovej jednotky sa riešia virtuálnou dedičnosťou. Každý si definuje to, čo potrebuje, zachovávajúc jednotný interface.

```
// ... // časť z HQuarters.h
class HeadQuarters {
public:
    HeadQuarters ();
    virtual ~HeadQuarters ();
    virtual void Birth () abstract;
    virtual void UpdateAll () abstract;
    virtual void Reset () abstract;

    virtual void Save ( const char* filename ) abstract;
    virtual bool Load ( const char* filename ) abstract;

    void ResetObj ();

    virtual vector< string > Info () abstract;

    int GiveNearest ( BaseAnt Ant );
    void IncFoodIn ();

    unsigned int Get_AntsOut () const { return m_AntsOut; }
    unsigned int Get_FoodIn () const { return m_FoodIn; }
    int Get_Cycle () const { return m_Cycle; }
    vector< int > Get_FoodNums () const { return m_FoodNums; }
    vector< double > Get_AvFits () const
    { return m_AverageFitness; }
    vector< double > Get_BestFits () const { return m_BestFitness; }

    vector< BaseAnt > m_Ants;
    vector< Food > m_Food;
    vector< Pheromone > m_Pheromone;

protected:
    unsigned int m_AntsOut;
    unsigned int m_FoodIn;
```

```

int                m_Cycle;

vector< int >       m_FoodNums;
vector< double >    m_AverageFitness;
vector< double >    m_BestFitness;
};
// ...

```

Na začiatku behu aplikácie (alebo pri načítaní konfiguračného súboru) sa skonštruuje a nainicializuje HeadQuarters – riadiaca jednotka. Do pointeru na základnú triedu sa priradí na novo vzniknutý potomok, čím sa vytvorí priestor pre virtuálnu dedičnosť.

Základná inicializačná časť je pre každý typ rovnaká. Rezervujú sa zdroje, spočítajú sa počty potravy, nulujú sa premenné. Ďalšie spoločné metódy sú: GiveNearest – vráti najbližší daný objekt pre jedinca (využíva sa k zisťovaniu kolízií, skúmaní vzájomného vzťahu s potravou alebo mraveniskom), IncFoodIn – zvyšuje čítač dovezenej potravy, ResetObj – znovunastavenie niektorých objektov v prostredí.

Ďalšie rysy centier sú už špecifickejšie, majú zdedenú len konvenciu volania.

- Riadiace centrum pre **riadenie automatom** (HQuartersAuto.cpp, HQuartersAuto.h):

Metóda Birth prichádza po spoločnej inicializačnej časti. Tá montuje do mravcov požadovaný typ mozgovej jednotky. Tak isto inšancuje potravu, poprípade i feromón.

UpdateAll má na starosti celý update v rámci jedného cyklu na najvyššej úrovni. Keď sa už dosiahol koniec života generácie, spočíta najlepších, priemerný fitness, počty dovezenej potravy, atď – napr. pre následné spracovanie Chartrom. Následne resetuje prostredie, mravce, a spúšťa novú generáciu. Keď sa ešte táto hranica nedosiahla, púšťa postupne mravce von z mraveniska, a káže im, aby sa aj oni aktualizovali. Tiež má na starosti evaporáciu feromónu v prostredí (ak je ovšem aktivovaná) – vyprchanie feromónu. Časom sila feromónu slabne, až kým úplne nezmizne. Takto sa nepoužívané cesty stratia, zostanú len tie, ktoré sú intenzívne využívané.

Metóda Info má na starosti výpis informácií na obrazovku v závislosti od typu riadiaceho centra – každý typ dáva iné údaje na obrazovku.

- Riadiace centrum pre **riadenie neurónovou sieťou** (HQuartersNN.cpp, HQuartersNN.h):

Tento typ mozgu je podobný tomu predchádzajúcemu, no má niekoľko charakteristík navyše. Už inicilizačná časť je o niečo zložitejšia, nakoľko je potrebné vytvárať chromozómy, celé prostredie genetického algoritmu a konštruovať neurónovú sieť.

V UpdateAll je tiež značná zmena, keďže je potrebné pri vzniku novej generácie spustiť genetický algoritmus, ktorý beží hore popísaným spôsobom. Modul genetickej zmeny je požiadaný o novú epochu, následne sa vyrobia nové (možno lepšie) mozgy, ktoré sa montujú do mravcov. Ostatné záležitosti sú analogické tým z prostredia automatov (evaporácia, štatistiky, reset, atď.).

1.2 Vizualizačná vrstva

(OGLInit.cpp, OGLInit.h; DrawEng.cpp, DrawEng.h)

Je realizovaná pomocou knihovne OpenGL (Open Graphics Library).

V inicializačnej časti sa vytvárajú fonty z bitmáp, z textúr, skonštruujú sa display listy kvôli rýchlejšiemu vykresľovaniu. Ďalej sa ešte nastavuje prostredie OpenGL (ShadeModel, hĺbkový buffer, antialiasing, blending, atď.)

```
// ... // časť z OGLInit.h
int InitGL ( GLvoid );
// nacistanie bitmap a konvertovanie na textury
int LoadGLTextures ();
// nahranie bitmapy
AUX_RGBImageRec *LoadBMP ( char *Filename );
// vytvorenie display listu pre fonty ( z textury )
GLvoid BuildFontText ( GLvoid );
// vytvorenie display listu pre fonty ( z bitmapy )
GLvoid BuildFontBitmap ( GLvoid );
// uvolnenie pameti fontov ( textura )
GLvoid KillFontText ( GLvoid );
// uvolnenie pameti fontov ( bitmapa )
GLvoid KillFontBitmap ( GLvoid );
// ...
```

Samotná vizualizácia sa odohráva v DrawScene (DrawEng.cpp, DrawEng.h). V jednotlivých cykloch prekresľuje dianie v prostredí mravčieho spoločenstva.

Na začiatku si prichystá plochu (zmazanie obrazovky, hĺbkového bufferu, napojenie textúry, pozadia, reset transformačnej matice, antialiasing). Kreslia sa postupne: okno prostredia, nadpisy, labely, informačné výpisy – v závislosti na type riadiaceho centra, riešené cez dedičnosť a virtuálne funkcie.

Pre následné zobrazovanie existujú dve možnosti. Podľa toho, či sme v grafovom režime, alebo či sa nachádzame režime znázorňujúcom priebeh simulácie.

Grafový režim:

(Charter.cpp, Charter.h)

Grafické znázornenie úspešnosti kolónie – pre budúce porovnanie rôznych mechanizmov riadenia hmyzu a pre pozorovanie vplyvov zmien parametrov a prostredia. To je možné robiť buď za behu, alebo pri pozastavenej simulácii.

Kreslenie je adaptívne, čiže ak beží simulácia v pozadí, tak sa menia aj rozsahy hodnôt, tým pádom aj celý graf. Cez konf. súbor sa dajú meniť max. hodnoty na osách, resp. krok značenia.

Grafový režim je celkom obecný, tvorí samostatný modul – trieda Chart. Sám si nastaví prostredie (metoda DrawBase), potom mu už stačí len odovzdať (metóda Plot a Hist) požadované údaje na vykreslenie. Sú dva spôsoby kreslenia priebehu: jeden ako graf a druhý podobný histogramu – hodí sa napr. na znázornenie vývoja počtu dovezenej potravy do mraveniska – de facto parameter úspešnosti spoločenstva.

Môže existovať viacero inštancií vykreslovača, podľa toho, koľko grafov resp. aké typy grafov chceme naraz znázorniť. Zadá sa len požadovaná poloha, o ostatné sa postará kresliaci modul (rozvrhne si rozloženie, škáluje podľa rozmerov okna, adaptuje rozsah osí apod.)

```

// ... // časť z Charter.h
class Chart {
public:
    Chart ( double oX, double oY, double axLenX, double axLenY );
    // skaluje, kresli vodiace linky
    void DrawBaseDbl ( int gener,
                      vector< vector< double > > toPlot );
    void DrawBaseInt ( int gener, vector< vector< int > > toHist );
    // kresli samotný graf
    void Plot ( vector< double > values, int rgb[3] ) const;
    // znázorňuje výsledky ala histogram
    void Hist ( vector< int > values, int rgb[3] ) const;
    // uloží znázornený graf do súboru
    bool SaveScreen ( unsigned ol, unsigned ot, unsigned sw,
                     unsigned sh, const char* filename ) const;

private:
    // kresli osy grafu
    void DrawAx ( ) const;
    // kresli popisky ku grafu
    void DrawLab ( ) const;

    double m_MaxX, m_MaxY;
    double m_OrigoX, m_OrigoY;
    double m_AxeXlen, m_AxeYlen;
};
// ...

```

Chart obsahuje metódu SaveScreen, ktorá dáva možnosť uložiť si výsledky jednotlivých simulácií vo forme obrázkov zo znázornených grafových priebehov.

Na uloženie sa používa TGA formát. V menu sa vyvoláva položkou Chart/Save As... (je aktívna iba v grafovom režime), po stlačení ktorej je možnosť si vybrať umiestnenie a meno súboru, do ktorého sa majú uložiť štatistiky na neskôr. Metóda načítá data z požadovanej pozície, alokuje pamäť, zisťuje formáty dát a následne posiela data na uloženie v TGA formáte.

Support pre samotný zápis do súboru vo formáte TGA (TgaSave.cpp, TgaSave.h):

```

// ... // časť z TgaSave.cpp
// hlavička TGA
#pragma pack( push, 1 )
struct TGAHeader
{
    unsigned char m_iIdentificationFieldSize;
    // colormap: 0-nie, 1-ano
    unsigned char m_iColourMapType;
    // RGB/RGBA/8bit - 2, colormap - 1
    unsigned char m_iImageTypeCode;
    // ignore
    unsigned short m_iColorMapOrigin;
    // veľkosť colormap
    unsigned short m_iColorMapLength;
    // bits per pixel colormap položky
    unsigned char m_iColourMapEntrySize;
    // ignore
    unsigned short m_iX_Origin;
    // ignore
    unsigned short m_iY_Origin;
    // šírka obrazku
    unsigned short m_iWidth;
    // výška obrazku
    unsigned short m_iHeight;
    // bits per pixel obrazku 8,16,24 alebo 32
    unsigned char m_iBPP;
};

```

```

        // ignore
        unsigned char m_ImageDescriptorByte;
    };

#pragma pack( pop )
// ...

```

Najprv sa vyplní hlavička súboru TGA všetkými potrebnými údajmi, následne prichádza na rad datová časť – takto vytvorený súbor sa potom uloží na disk.

Druhý režim – znázorňujúci priebeh simulácie:

Postupne sa kreslia prvky prostredia, potrava, mravenisko, feromón – farbí sa podľa intenzity – a napokon mravce. Všetky objekty prostredia sú poskladané z polygónov alebo samostatných čiar, tie sa vykresľujú rôznymi spôsobmi, ako napr. smyčka liniek, nezávislé linky, celé obdĺžniky a ďalšie útvary. Jeden z jednoduchších útvarov prostredia, ktorý je tiež komponentom zložitejších:

```

// ... // časť z DrawEng.cpp
void DrawObj ( double posX, double posY, double size )
{
    // reset matice
    glLoadIdentity();
    // posun
    glTranslated( posX, posY, 0.0f );
    glBegin( GL_LINE_LOOP );
        glVertex2d( 1.6*size, 0.0*size );
        glVertex2d( 1.2*size, 1.2*size );
        glVertex2d( 0.0*size, 1.6*size );
        glVertex2d( -1.2*size, 1.2*size );
        glVertex2d( -1.6*size, 0.0*size );
        glVertex2d( -1.2*size, -1.2*size );
        glVertex2d( 0.0*size, -1.6*size );
        glVertex2d( 1.2*size, -1.2*size );
        glVertex2d( 1.6*size, 0.0*size );
    glEnd();
}
// ...

```

Vizualizácia mravcov prebieha podobne, až na to, že majú zložitejšiu štruktúru. Časť kódu, ktorý túto úlohu vykonáva:

```

// ... // časť z DrawEng.cpp
void DrawAnt ( BaseAnt Ant )
{
    glLineWidth( 1.0f );
    double s = Ant.Get_Size();
    // reset matice
    glLoadIdentity();
    // posun
    glTranslated( Ant.Get_PosX(), Ant.Get_PosY(), 0.0f );
    // rotácia
    glRotated( RADTODEG( Ant.Get_Rotation() ), 0.0f, 0.0f, 1.0f );

    glColor3ubv( Ant.m_Color );
    glBegin( GL_LINES );
        glVertex2d( -0.5*s, 2.75*s );
        glVertex2d( -0.25*s, 1.0 *s );
        ...
    glEnd();
}
// ...

```

Niektoré objekty sa kreslia pomocou display listu – sú to predvytvorené objekty, pomocou ktorých sa všetko vytvorí len raz. Keďže je objekt v pamäti (v display liste), OpenGL nemusí všetko prepočítavať. Odstráni sa tak značné zaťaženie procesoru.

```
// ... // časť z DrawEng.h
int DrawGLScene ( Headquarters* HQ );
GLvoid glPrintText ( GLint x, GLint y, int set,
                    const char *fmt, ... );
GLvoid glPrintBitmap ( const char *fmt, ... );

void DrawObj ( double PosX, double PosY, double Size );
void DrawSmallObj ( double PosX, double PosY, double Size );
void DrawFood ( Food Fd );
void DrawAnt ( BaseAnt Ant );
void DrawAntWithFood ( BaseAnt Ant );
// ...
```

2 Beh programu

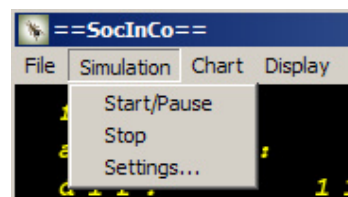
Po spustení aplikácie sa automaticky začne simulácia spoločenstva s dopredu nastavenými parametrami. Tie je možné za behu meniť dvoma spôsobmi: cez konfiguračný súbor a pomocou dialógu. Toto dvojúrovňové zadávanie parametrov oddeľuje „užívateľské parametre“ od „pokročilejších“. Dialógom je možné nastavovať parametre slúžiace užívateľovi, ako napr. typ riadiacej jednotky, počet mravcov v prostredí, počet potravy, jej min. vzdialenosť od mraveniska, či sa má používať feromón, či má feromón vyprchávať, atď. „Programátorské parametre“ sa nastavujú v konfiguračnom súbore, menia povahu algoritmov, aplikácie. Napr. senzibilita mravcov, ich fyzické vlastnosti, veľkosti okna, prostredia, náповedy v dialógovom zadávaní parametrov a mnoho ďalších v závislosti tiež na type riadenia (napr. u neurónových sietí: pomer mutácie, kríženia, počet neurónov, vrstiev, prahy, atď.).

Spôsob zmeny parametrov, ako aj jednotlivé možnosti, sú popísané v užívateľskej dokumentácii, v sekcii ovládania aplikácie.

3 Ovládanie aplikácie

Menu - Slúži na riadenie simulácie, na riadenie samotného programu. Cez neho je možné dostať sa k nastaveniam, prepínať medzi rôznymi zobrazeniami, dostať sa k výsledkom behov.

Popis jednotlivých položiek sa nachádza v užívateľskej dokumentácii.



Položky menu:

File:

- Load Params
- Load
- Save
- Exit

Simulation:

- Start/Pause
- Stop
- Settings...

Chart:

- Show
- Over
- Save As...

Display:

- Display/Pheomone
- Display/Details
- Display/Fullscreen
- Display/Antialiasing

SocInCo – Užívateľská dokumentácia

Autor: Attila Ulman (ulman.attila@gmail.com)
Vedúci práce: RNDr. Pavel Surynek
Názov: Vznik inteligentného chovania u hmyzích
spoločenstiev



1 Motivácia

Práca sa zaoberá skúmaním a simuláciou rozsiahleho spoločenstva jednoduchých agentov – mravcov. Títo relatívne jednoduchí jedinci dokážu v spojení dosiahnuť oveľa zložitejšie ciele, ktoré viacnásobne prevýšia ich individuálne schopnosti. Sú schopní kolektívne riešiť úlohy, ktoré by jednotlivci nemali šancu zvládnuť a to bez existencie nejakej riadiacej autority a dopredu daných fixných štruktúr. Aplikáciou jednoduchých pravidiel vzniká komplexné chovanie celku. Zaujímavý je najmä ich primárny životný cieľ – hľadanie potravy – práve tu sa najviac prejavuje ich kooperácia a inteligencia celého systému ako „superorganizmu“, „inteligentnej entity“. Kolónia si veľmi rýchlo vie nájsť cestu k potrave pomocou feromónových značiek. Pri pohybe od mraveniska k zdroju potravy a späť každý mravec ukladá chemickú látku zvanú feromón. Ostatné mravce sú schopné detekovať feromón a aj rozlišovať jeho rôzne koncentrácie a podľa najvyššej si vyberajú cestu k potrave. Čím viac mravcov používa danú cestu, tým väčšia bude jej koncentrácia feromónu, kým ostatné nepoužívané cesty pomaly vyprchajú. Žiadna prekážka nedokáže tomuto putovaniu zabrániť, mravce si vždy nájdu ďalšiu najlepšiu cestu. Týmto vlastne vytvárajú dynamicky sa meniacu štruktúru medzi nimi (mraveniskom) a potravou, ktorá je schopná prispôbovať sa meniacim podmienkam. Vlastnosti tohto typu sa s obľubou čoraz častejšie používajú pre optimalizačné úlohy (Ant Colony Optimization) a pre aplikácie v sieťovom prostredí (routing).

Dá sa povedať, že mravce komunikujú cez prostredie: Aktivita jedincov má za výsledok feromónovú stopu, ktorú mravce nasledujú, čiže prostredie vplýva na ich aktivitu. Týmto vzniká nepriama komunikácia medzi jedincami. Samozrejme je tu ešte mnoho vplyvov, ako napr. zdroj svetla, vlhkosť, prekážky, samotné vlastnosti jedince, atď.

2 Popis projektu – aplikácie

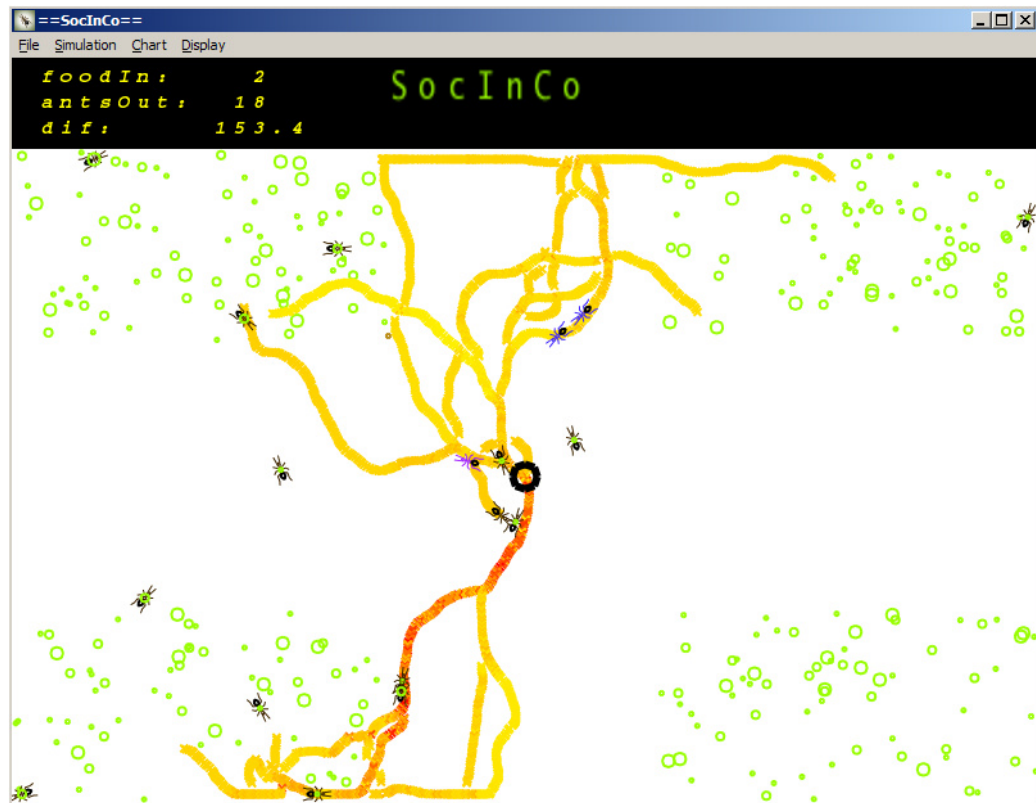
Aplikácia SocInCo je súčasťou Bakalárskej práce. Ide o výskumnú prácu, cieľom ktorej je skúmať chovanie sociálneho hmyzu, jeho vznik, pozadie a súvislosti a pokúsiť sa vytvoriť simuláciu „in silico“ ktorá je čo najbližšie k realite a čo najviac zachycuje inteligentnú kooperáciu jedincov. Pokúsiť sa vytvoriť inteligentné chovanie kolónie a dosiahnuť zaujímavé teoretické výsledky. Pomocou simulácie pochopiť, ako môžu napr. mravce kolektívne uspieť, prekonať tak svoje individuálne obmedzené schopnosti. Taktiež pochopiť emergentné schopnosti mravčích kolónií, ktoré môžu byť nástrojom štúdia kolektívneho chovania vyspelejších spoločenskostí.

Ďalším významným cieľom je porovnať rôzne prístupy riadenia jedincov a skúmať, ktorá metóda je najviac realistická resp. najefektívnejšia (čo sa týka putovania za potravou). Pri dokonalosti prírody tieto dva aspekty majú k sebe veľmi blízko. Skúmať rôzne analógie s reálnou prírodou a tiež vplyv rôznych zmien vstupných parametrov simulácie. Vyvodit' závery jednak z porovnania prístupov riadenia, algoritmov ako aj z rôznych zmien parametrov, prostredia, vlastností celého systému. Dôraz sa kladie na čo najvernejšiu podobu prírodných javov a na samotné vytvorenie inteligentného chovania.

Simulačný systém – popisovaná aplikácia (SocInCo) – umožňuje študovať vznik inteligentného chovania hmyzích spoločenskostí a porovnávať rôzne prístupy. Jej súčasťou je vizualizácia simulácie, vyhodnocovanie simulácie, porovnávanie, grafické znázornenie vybraných parametrov, úspešnosti kolónie, ich uloženie, zadávanie parametrov simulácie, prostredia. Obsahuje dve metódy riadenia jedincov: konečný automat a neurónovú sieť v kombinácii s genetickými algoritmami.

3 Prostredie simulácie

Mravce sú situované do 2D ohraničeného prostredia, z ktorého nemôžu vypadnúť ani ho opustiť. Mravce nekolidujú s ostatnými, ale preliezajú cez seba. Tento jav nie je cudzí od prírody, kde mravce tak isto lezú po sebe, napr. mnohokrát vytvárajú takto mravčie „mosty“, alebo sa s tým môžeme stretnúť aj pri budovaní mravenísk. Prostredie obsahuje potravu viacerých veľkostí, ktoré je rozmiestňovaná náhodne. Jednotlivé počty potravy sú nastaviteľné užívateľom. Potrava sa samozrejme generuje v určitej vzdialenosti od mraveniska, aby sa prejavilo putovanie za potravou (nastaviteľná vzdialenosť). Táto metóda simuluje viac realistické hľadanie ciest k zdroju potravy. Keď sa mravec dostane do kolízie s potravou, vezme potravu so sebou a pokúsi sa ju odnieť do mraveniska.



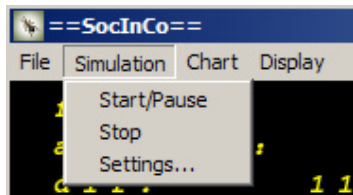
Screenshot prostredia

4 Beh programu

Po spustení aplikácie sa automaticky začne simulácia spoločenstva s dopredu nastavenými parametrami. Tie je možné za behu meniť dvomi spôsobmi: cez konfiguračný súbor a pomocou dialógu. Toto dvojúrovňové zadávanie parametrov oddeľuje „užívateľské parametre“ od „pokročilejších“. Dialógom je možné nastavovať parametre slúžiace užívateľovi, ako napr. typ riadiacej jednotky, počet mravcov v prostredí, počet potravy, jej min. vzdialenosť od mraveniska, či sa má používať feromón, či má feromón vyprchávať, atď. „Programátorské parametre“ sa nastavujú v konfiguračnom súbore, menia povahu algoritmov, aplikácie. Napr. senzibilita mravcov, ich fyzické vlastnosti, veľkosti okna, prostredia, nápovedy v dialógovom zadávaní parametrov a mnoho ďalších v závislosti tiež na type riadenia (napr. u neurónových sietí: pomer mutácie, kríženia, počet neurónov, vrstiev, prahy, atď.)

Spôsob zmeny parametrov, ako aj jednotlivé možnosti, sú popísané nižšie, v sekcii ovládania aplikácie.

5 Ovládanie aplikácie

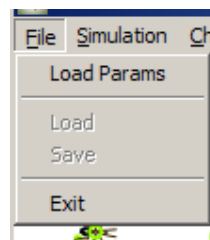


Menu - Slúži na riadenie simulácie, na riadenie samotného programu. Cez neho je možné dostať sa k nastaveniam, prepínať medzi rôznymi zobrazeniami, získať výsledky behov.

Položka File:

- Load Params

Možnosť meniť „programátorské“ (ale aj „užívateľské“) parametre cez konfiguračný súbor. Po stlačení sa získajú nastavenia zo súboru. Potom sa simulácia, ako aj celé okno resetne (to napr. kvôli zmenám rozmerov a pod.)



Zadávanie znesie pomerne veľa. Keď sa zadá menej parametrov, ako je možné zadať, tak sa nastaví len tie, ostatné získajú defaultné hodnoty. Naopak, keď sa zadá viac parametrov, pochopiteľne sa nerozoznajú a všetky nastavenia budú východzie. Podobná je situácia i v prípade, že nastane nejaká iná chyba. (napr. chýbajúci súbor, nesprávny formát, sporné údaje, atď.)

Akákoľvek neštandardná situácia je hlásená spolu s číslom riadku, kde nastala.

Na poradi parametrov v súbore nezáleží, biele znaky a komentáre (// do konca riadku) sa preskakujú.

```
NestSize = 6
// neural:
Bias = -1
Perturbation = 0.3
NBestEliteAnts = 4
```

Mená parametrov sú samovysvetľujúce, ich zoznam je nasledujúci:

programátorské:

```
FoodTypes, FoodSize, AntSize, PherSize, AntsNumMin,
AntsNumMax, FoodNumMax, MinNestDistMax,
MaxTurnTolerance, MinTurnTolerance, Title,
WindowSizeX, WindowSizeY, LeftMargin, RightMargin,
UpperMargin, LowerMargin, NestPosX, NestPosY,
NestSize, Bias, Perturbation, NBestEliteAnts,
HiddenLayersNum, NeuronsPerLayer, InputsPerNeuron,
NeuronOutputsNum, MutRate, CrossRate, ReturnReward,
FoundReward, MaxX, MaxY.
```

užívateľské:

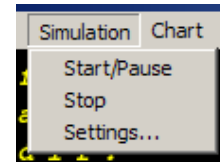
```
AntsNum, FoodNum, MinNestDist, GenerationLifeTime,
BrainType, UsePher, EvaporPher.
```

Tam, kde sa očakáva „pole“ hodnôt, je potrebné písať jednotlivé hodnoty za sebou, oddelené medzerami alebo tabulátormi (ľub. počtom)

- **Load**
Možnosť načítať riadiacu jednotku – mozog (aktivované iba vtedy, ak to má zmysel, napr. evolvovaná neurónová sieť)
- **Save**
Možnosť uložiť riadiacu jednotku – mozog (aktivované iba vtedy, ak to má zmysel, napr. evolvovaná neurónová sieť)
- **Exit**
Ukončí beh programu.

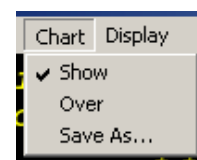
Položka Simulation:

- **Start/Pause**
Riadenie simulácie. Je možné pozastaviť simuláciu, keď je potrebné niečo presnejšie pozorovať, alebo pri kreslení grafu, aby bol k dispozícii statický obraz.
- **Stop**
Zastaví simuláciu úplne. Po stlačení **Start/Pause** sa vytvorí všetko nanovo a môže všetko začať odznova.
- **Settings...**
Vyvolá dialógové okno, kde je možné nastavovať rôzne parametre ovplyvňujúce ako prostredie tak aj mravce. Pokročilejšie zásahy je možné robiť cez vyššie zmieňovaný konfiguračný súbor.
Dovoľuje vymieňať typy mozgov, zadávať počty mravcov v prostredí, počty rôznych typov potravy, ako aj ich minimálnu vzdialenosť, v ktorej sa generujú. Pomocou neho sa dá aktivovať/deaktivovať aj používanie feromónu mravcami (je zaujímavé sledovať úspešnosť populácie pri zapnutom/vypnutom feromóne), ako aj jeho prípadnú evaporáciu.
Pri zmene niektorého z parametrov sa všetko vytvorí nanovo a začne sa nová simulácia. Všetky hranice sa kontrolujú, nápovedy k možným hodnotám sa generujú z „programátorských“ parametrov. (konf. súbor)



Položka Chart:

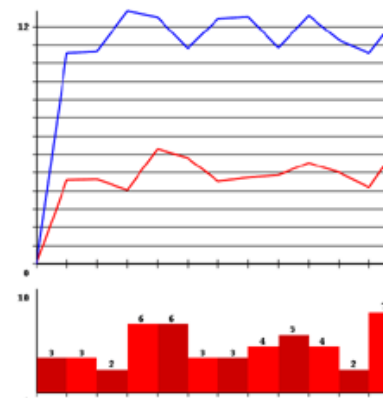
Grafické znázornenie úspešnosti kolónie – pre porovnanie rôznych mechanizmov riadenia hmyzu a pre pozorovanie vplyvov zmien parametrov a prostredia. Existujú dva typy znázornenia: jeden grafový a druhý podobný histogramu (hodí sa na znázornenie vývoja počtu dovezenej potravy do mraveniska).



- **Show**

Možnosť prepnúť sa do grafového režimu, kde je možné sledovať vývoj daných špecifik v čase. (priem. a max. fitness, počty dovezenej potravy)

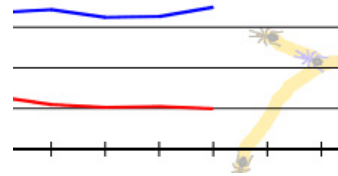
To je možné robiť buď za behu, alebo pri pozastavenej simulácii. Kreslenie je adaptívne, čiže ak beží simulácia v pozadí, tak sa menia aj rozsahy hodnôt, tým pádom aj celý graf. Cez konf. súbor



sa dajú meniť max. hodnoty na osiach, resp. krok značenia.

- Over

Spôsobí, že sa vyhodnotenie výsledkov, grafy priebehov znázorňujú „cez“ prebiehajúcu simuláciu, pomocou čoho je možné naraz sledovať prejavy modelu (v nie tak intenzívnom prevedení) spolu s číselnými a grafickými výsledkami.



Položka je aktívna iba v grafovom režime, jej stav sa zapamätá.

- Save As...

Možnosť uložiť si výsledky jednotlivých simulácií vo forme obrázkov zo znázornených grafových priebehov. Na uloženie sa používa TGA formát.

Položka je aktívna iba v grafovom režime. Po jej stlačení je možné vybrať si umiestnenie a meno súboru, do ktorého si chceme uložiť štatistiky na neskôr.

Položka Display:

Rôzne spôsoby zobrazenia.

- Display/Pheomone

Zapína vykresľovanie feromónu.

- Display/Details

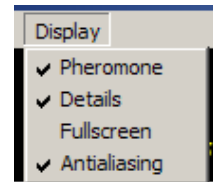
Dovoľuje vypisovať ku každému jedincovi jeho detaily (napr. hodnotu fitness)

- Display/Fullscreen

Beh vo fullscreen.

- Display/Antialiasing

Zapína vykresľovanie s antialiasom.



6 Ako to funguje v prírode...



Mravce nesúce lupene kvetov



Intenzívne využívané feromónové cesty